ControlLogix SIL2 System Configuration

Using RSLogix 5000 Subroutines

Application Technique

(Catalog Numbers 1756 and 1492)

ALLEN-BRADLEY • ROCKWELL SOFTWARE

Rockwell Automation

KLINKMANN
www.klinkmann.com

Important User Information

Solid state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://literature.rockwellautomation.com) describes some important differences between solid state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.

WARNING	Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.
IMPORTANT	Identifies information that is critical for successful application and understanding of the product.
ATTENTION	Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence
SHOCK HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.
BURN HAZARD	Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.

Allen-Bradley, ControlLogix, TechConnect, RSLogix 5000, RSNetWorx for ControlNet, Rockwell Automation, and RSLinx are trademarks of Rockwell Automation, Inc.

Trademarks not belonging to Rockwell Automation are property of their respective companies.

Updated Information

Revision **B** of this publication contains the new or updated information listed in this table.

New or Updated Information in This Publication

Description	Chapter	Pages
Software and program requirements for the fault-tolerant system.	Chapter 1	21
Enhanced descriptions of system states and added graphics.	Chapter 3	52…55
Updated graphics for consistency with the most-recent version of the SIL2_IO_Fault_Tolerant program.	Chapter 4	65…103
Call_Code subroutine JSR parameters - additional input parameters for each module pair are shown and described.	Chapter 4	85…103
Programming for a demand - examples updated.	Chapter 5	105…116
Added information about 1756-IB32 module replacement.	Chapter 6	117…130
Appendix of frequently-asked-questions added.	Chapter D	155…162
Corrections to topics and page number references.	Index	167…163

New or updated information in this manual is indicated with a change bar as seen to the right of this paragraph, except for changes to the index.

Preface

About This Publication 11
Who Should Use This Publication 11
Conventions 11
About SIL 11
Additional Resources 12

Chapter 1

The Fault-tolerant System Configuration

About This Chapter 13
Fault Tolerance and ControlLogix 13
ControlLogix System SIL2 Configurations 13
About Fault-tolerant Systems 14
Fault-tolerant Compared to Other SIL2 Configurations . . . 14
Fault-tolerant System Configuration 16
Remote I/O Configuration 16
The Complete ControlLogix Fault-tolerant System 20
Hardware 20
Software and Programming 21
Additional Resources 22

Chapter 2

Fault-tolerant System Hardware

About This Chapter 25
Approved I/O Modules and Termination Boards 25
About the Specialized Termination Boards 26
1756-IB32 DC Input Termination Board Features 26
Normal Operation of 1756-IB32, DC Input Termination Board 27
1756-IB32 DC Input Termination Board and Transition Tests 28
1756-IF16 Analog Input Termination Board 30
Normal Operation of the 1756-IF16, Analog Input Termination Board 31
One-sensor or Two-sensor Wiring Option 33
1756-IF16 Module Pair Reference Tests 34
1756-OB16D Diagnostic Output Termination Board Features 37
Normal Operation of the 1756-OB16D Diagnostic Output Termination Board 38
Diagnostic Tests and the 1756-OB16D Output Termination Board 39
Termination Board Relay Control 40
1756-IB32 Input Termination Board Relay Control 40
1756-IF16 Analog Input Termination Board Switch Control 41
1756-OB16D Output Termination Board Relay Control . . 42
Input Module Diagnostic Test Control 44
Hardware and Programming 44
Additional Resources 45

Fault-tolerant Program Elements

Chapter 3

About This Chapter 47
Overview of the Program Elements 47
Main Routine 47
Diagnostic Subroutines 48
Diagnostic Features of Subroutines 48
Call_Code Subroutines 49
Function of the Program Elements 50
Program Elements Provided 51
States of the System 52
Normal State 52
Test State 52
1oo1 State 53
Faulted State 54
IB32_Diagnostics Subroutine 55
Normal Operation - 1756-IB32 Module Pair 55
Test - 1756-IB32 Module Pair 56
1oo1 - 1756-IB32 Module Pair 56
IF16_Diagnostics Subroutine 57
Normal Operation - 1756-IF16 Module Pair 57
Test - 1756-IF16 Module Pair 58
1oo1 - 1756-IF16 Module Pair 58
IF16_RefCal Subroutine 59
OB16D_Diagnostics Subroutine 60
Normal Operation - 1756-OB16D 60
1oo1 - 1756-OB16D 61
Data Flow Between Program Elements 62
The Fault-tolerant Program 63
Additional Resources 63

Configuring the Fault-tolerant System

Chapter 4

About This Chapter 65
Before You Begin 65
Begin with the Fault-tolerant I/O Program 66
Adding a CNB or CNBR to the Controller Chassis 66
Configuring Remote I/O Chassis 67
Add the Remote I/O Chassis to the I/O Configuration Tree. 67
About System-generated Tags. 71
Specifying Diagnostic Subroutine Behavior. 72
About ModulePair Tags 72
Create ModulePair Tags 73
Edit ModulePair Tags 76
Editing 1756-IB32 ModulePair Tags. 77
Editing 1756-IF16 ModulePair Tags 79
Editing 1756-OB16D ModulePair Tags. 82
Adding MESSAGE Tags 84
Editing the Call_Code Subroutines 84
Editing the 1756-IB32 Call_Code Subroutine 85
Copy and Paste a JSR Rung for Each 1756-IB32 Module Pair 85
Edit JSR Parameters for the 1756-IB32 Module Pair 87
Edit Other Rung Elements for the 1756-IB32 Module Pair 88
Editing the 1756-IF16 Call_Code Subroutine 90
Copy and Paste a JSR Rung for Each 1756-IF16 Module Pair. 90
Edit JSR Parameters for the 1756-IF16 Module Pair 92
Edit Other Rung Elements for the 1756-IF16 Module Pair. 93
Editing the 1756-OB16D Call_Code Subroutine 95
Copy and Paste Rungs for Each 1756-OB16D Module Pair 95
Edit Elements of the 1756-OB16D Call_Code Routine . . . 97
Edit JSR Parameters for the 1756-OB16D Module Pair . . 102
Next Steps 103
Additional Resources. 103

Chapter 5

Programming the Fault-tolerant System

About This Chapter 105
Programming the Main Routine 105
Relationship Between Main Routine and Diagnostic Subroutines 106
Basic Input/Output Programming 106
.I and .O Data in Fault-tolerant Programming 106
Example Input/Output Rung 107
Module Pair Fault to Result in System Shutdown 108
Fault Reset Programming 109
Circuit Reset Programming 111
Circuit Reset Programming Considerations 111
Programming for a Demand on the System 113
Demand Made Through a 1756-IB32 Module Pair 113
Demand Made Through a 1756-IF16 Module Pair 114
Power-up Sequence 115
Additional Resources 116

Chapter 6

Troubleshooting a Fault-tolerant System

About This Chapter 117
Identifying a Faulted Module Pair 118
Example of Programming to Identify a Faulted Module Pair 120
Identifying a Faulted Module 121
Replacing a Faulted 1756-IB32 Module 121
1756-IB32 ModulePair Tags to Identify the Type of Module Fault 122
1756-IF16 ModulePair Tags to Identify the Type of Module Fault 123
1756-OB16D ModulePair Tags to Identify the Type of Module Fault 124
Using Resets 125
When to Use the Fault Reset 125
When to Use Circuit Reset 125
Examples of Faults and Resulting Tag Values 126
1756-IB32 Module Pair - One Module Faulted 126
1756-IF16 Module Pair - One Module Faulted and Removed 127
1756-IF16 Module Pair - Two Modules Faulted 128
Additional Resources 129

SIL2 Remote I/O Fault-tolerance Tags

Appendix A

About This Appendix 131
1756-IB32 ModulePair Tags 131
1756-IB32 ModulePair Tags for System Behavior 131
1756-IB32 Module Status Tags 133
1756-IB32 ModulePair Tags for Use in Programming . . . 135
1756-IB32 Hidden Tags, Not for Use 136
1756-IF16 ModulePair Tags 137
1756-IF16 ModulePair Tags for System Behavior 137
1756-IF16 Module Status Tags 138
1756-IF16 ModulePair Tags for Use in Programming . . . 141
1756-IF16 Hidden Tags, Not for Use 142
1756-OB16D Module Pair Tags 143
1756-OB16D ModulePair Tags for System Behavior 143
1756-OB16D Module Status Tags 144
1756-OB16D ModulePair Tags for Use in Programming . 146
1756-OB16D Hidden Tags, Not for Use 147

SIL2 Fault-tolerant Topology

Appendix B

About This Appendix 149
Planning Considerations 149

Fault-tolerant System Limitations

Appendix C

About This Appendix 153
About Faults and Overall Fault-tolerance 153
Detecting System-side Versus Field-side Faults 153
Limits of Fault-detection from the 1756-OB16D Termination Board 153
Module Pair Faults 154

Frequently Asked Questions

Appendix D

About This Appendix 155
About Redundant Chassis 155
About I/O 157
About Fail-safe and Fault-tolerant Programs 160

Glossary

Index

About This Publication

This publication provides techniques and guidelines for configuring a SIL2-certified, ControlLogix fault-tolerant system. This publication provides only recommendations for how to configure a fault-tolerant system for SIL2 compliance and is not a comprehensive reference of ControlLogix SIL2 information.

Other publications and resources outlined in the Additional Resources table on page 12 should also be consulted and used as references when configuring a ControlLogix SIL2 safety application.

Who Should Use This Publication

This publication is intended for use only by individuals who have extensive knowledge of safety applications, SIL policies, programmable control systems, and ControlLogix products. Do not use this publication if you do not fully understand these concepts.

Conventions

The following writing conventions are used in this publication.

Text that is	Identifies
Italic	A variable that you replace with your own text or value
`courier`	Example programming code, shown in a monospace font so you can identify each character and space

In addition to the textual conventions described, note that underlined text, chapter title references, section title references, table title references, and page numbers function as hyperlinks in the electronic version of this publication.

About SIL

The International Electrotechnical Commision (IEC) has defined Safety Integrity Levels (SILs) in IEC publication 61508. Concepts and terms explained in this reference manual are based upon publication 61508.

A SIL is a level in the IEC rating system used to specify the safety integrity requirements of a safety-related control system. SIL1 is the lowest level and SIL4 is the highest. For more information about SIL specifications, see IEC publication 61508-1, General Requirements.

Additional Resources

The following resources should also be consulted when configuring a ControlLogix system for SIL2 certification.

Resource	Description
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.
ControlLogix Controllers User Manual, publication 1756-UM001	This manual explains the general use of ControlLogix controllers.
ControlLogix Redundancy System User Manual, publication 1756-UM523	This user manual explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Functional safety of electrical/electronic/programmable electronic safety-related systems, IEC 61508	IEC 61508 describes terms, component requirements, process requirements, and techniques for SIL2 applications.

The Fault-tolerant System Configuration

About This Chapter

This chapter explains how the fault-tolerant configuration differs from the fail-safe and high-availability configurations and provides a brief overview of the fault-tolerant configuration and application.

Topic	Page
Fault Tolerance and ControlLogix	13
ControlLogix System SIL2 Configurations	13
About Fault-tolerant Systems	14
Fault-tolerant Compared to Other SIL2 Configurations	14
Fault-tolerant System Configuration	16
Remote I/O Configuration	16
Additional Resources	22

Fault Tolerance and ControlLogix

This section briefly describes the newly-certified fault-tolerant configuration.

ControlLogix System SIL2 Configurations

The following ControlLogix system configurations are certified for use in SIL2 applications and are described further in the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001:

- Fail-safe
- High-availability
- Fault-tolerant

The fault-tolerant configuration is the most recent to be made available.

8 / 2011 www.klinkmann.com KLINKMANN

About Fault-tolerant Systems

IEC publication 61508-4 defines fault tolerance as the "ability of a functional unit to continue to perform a required function in the presence of faults or errors."

While not completely fault tolerant, the ControlLogix SIL2 system is described as fault tolerant because it is able to tolerate a majority of faults that may occur in the system. In the unlikely event of a fault where the safety system cannot carry-out the safety application, the system fails-to-safe.

For more information about the limits of the fault-tolerant system, see Fault-tolerant System Limitations, on page 153.

Fault-tolerant Compared to Other SIL2 Configurations

Other ControlLogix SIL2 configurations, fail-safe and high-availability, are not fault-tolerant.

Fail-safe Configuration

In the fail-safe system, if a fault occurs anywhere in the system (that is, in the controller, communications, or I/O) an Emergency Shutdown (ESD) occurs. The fail-safe configuration is further described in Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001 and is not shown here.

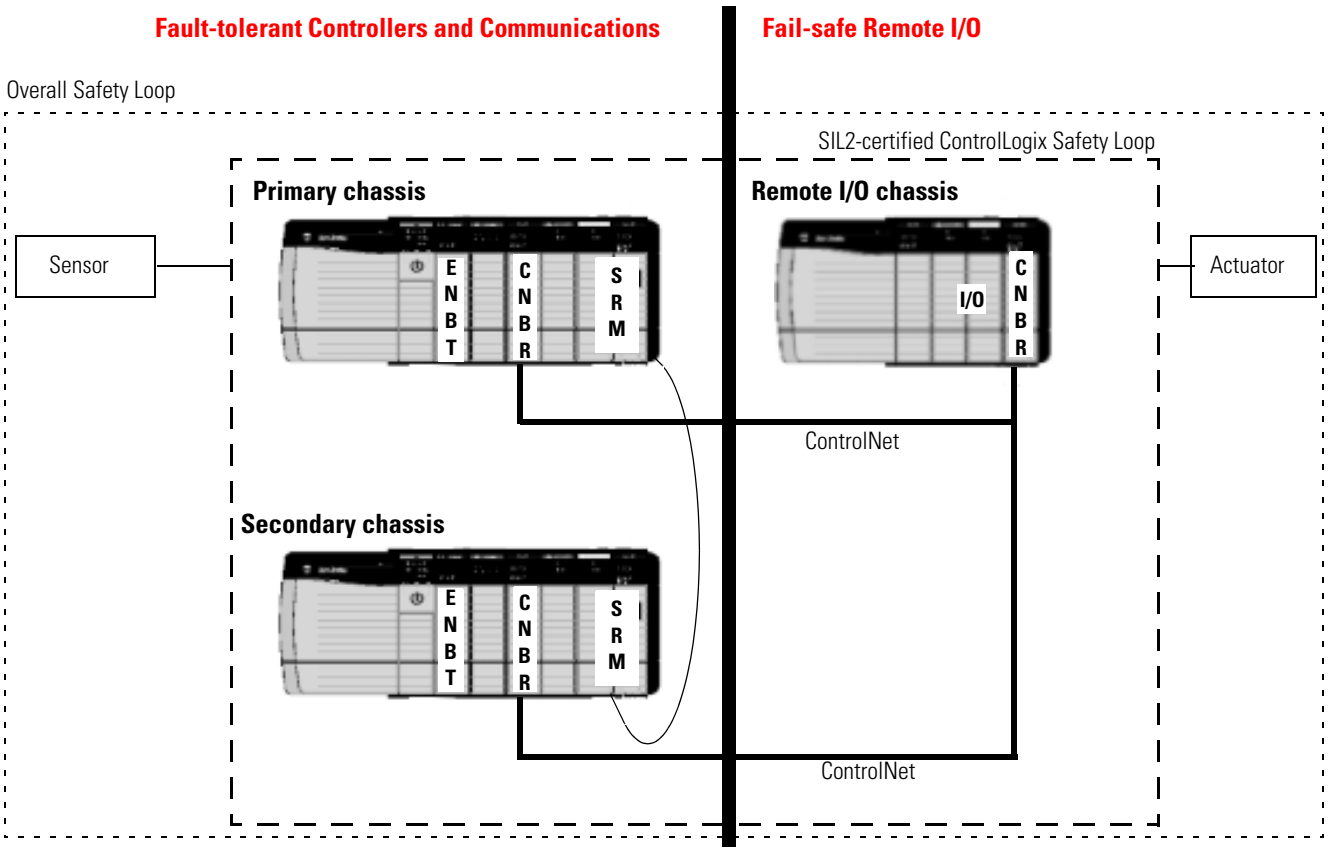
High-availability Configuration

In the high-availability configuration, the controller and communication chassis are fault tolerant, but the remote I/O is not. In the high-availability configuration, if a fault occurs in either the primary or secondary chassis, the system can continue to carry out the safety function. If a fault occurs in the remote I/O chassis of the high-availability configuration, the system fails to safe.

See the High-availability Configuration graphic for a depiction of the division between the fault tolerant and the fail safe portions of the high-availability configuration.

For example, if a fault occurs in the controller of the primary chassis, the safety system can continue to operate despite the fault. However, if a fault occurs in the remote I/O chassis (on the right side of the diagram), the system fails-to-safe.

High-availability Configuration



Fault-tolerant Configuration

The fault-tolerant configuration provides more fault tolerance than the high-availability configuration because remote I/O chassis are also configured to be fault tolerant.

Fault-tolerance in a SIL2-certified ControlLogix system is achieved by the use of redundant controller and communication chassis, redundant remote I/O chassis, specialized I/O termination boards, and special application programming.

Fault-tolerant System Configuration

The ControlLogix fault-tolerant system configuration uses some elements from the high-availability configuration and other elements that are specific only to the fault-tolerant configuration.

In a fault-tolerant configuration, the controller and communication chassis are configured as specified for the high-availability configuration (see the left side of High-availability Configuration graphic).

The fault-tolerant configuration differs from the high-availability configuration because of the remote I/O configuration.

Remote I/O Configuration

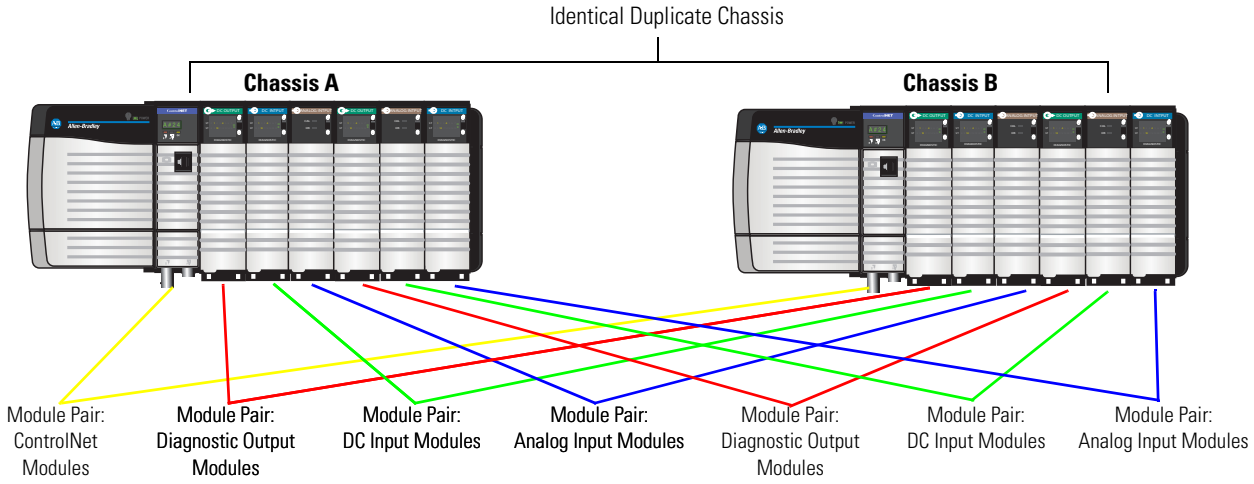
In a fault-tolerant configuration, the remote I/O chassis are configured in duplicate, identical pairs. The duplicate chassis must be identical in the modules used, as well as the location and configuration of the modules. Each I/O module in the chassis pair should have an exactly identical module in the same slot of the other chassis of the duplicate pair.

Your ControlLogix fault-tolerant system may use any number of identical, duplicate remote I/O chassis within the limits of your controller.

Within the identical, duplicate remote I/O chassis are the I/O modules certified for use in the SIL2 system. Because chassis are configured identically, each module in chassis A should have duplicate in chassis B. The duplicate I/O modules (one each chassis) are referred to as module pairs.

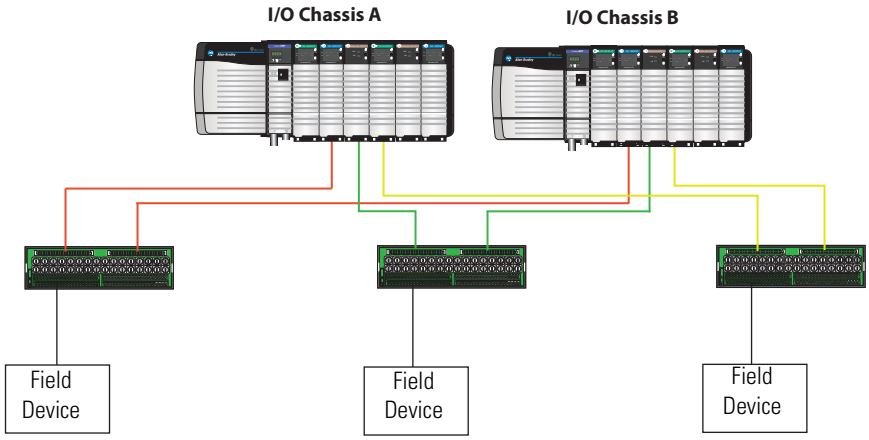
The concept of identical, duplicate remote I/O chassis is depicted in the graphic below. In this publication, the duplicate remote I/O chassis are identified by an uppercase letter. For example, Chassis A and Chassis B would indicate a duplicate remote I/O chassis pair.

Identical, Duplicate Remote I/O Chassis


Identical Duplicate Chassis
Chassis A
Chassis B
Module Pair: ControlNet Modules
Module Pair: Diagnostic Output Modules
Module Pair: DC Input Modules
Module Pair: Analog Input Modules
Module Pair: Diagnostic Output Modules
Module Pair: DC Input Modules
Module Pair: Analog Input Modules

In addition to the identical, duplicate remote I/O chassis, the fault-tolerant system also requires the use of specialized I/O termination boards. Each module pair is connected to a specialized termination board. Each termination board is wired to field devices such as sensors and actuators.

Remote I/O Chassis with Termination Boards



How Remote I/O Interacts with Termination Boards

The specialized termination boards have several functions related to remote I/O. The following are functions that all three types of termination boards provide.

- Simplified connections from field devices to like modules in both chassis of the duplicate remote I/O chassis.
- Electrical isolation to prevent module channels from interfering with each other.

In addition to the functions described above, functions specific to each type of I/O module are also provided. The following table identifies and describes I/O module-specific functions.

I/O Module-specific Functions

I/O Module Type	Function
Input module	Executes diagnostic tests initiated by the control program. The tests help the system verify that the input modules are working as expected.
Output module	On-board relays provide a secondary method of disconnect between the I/O modules and their power source.

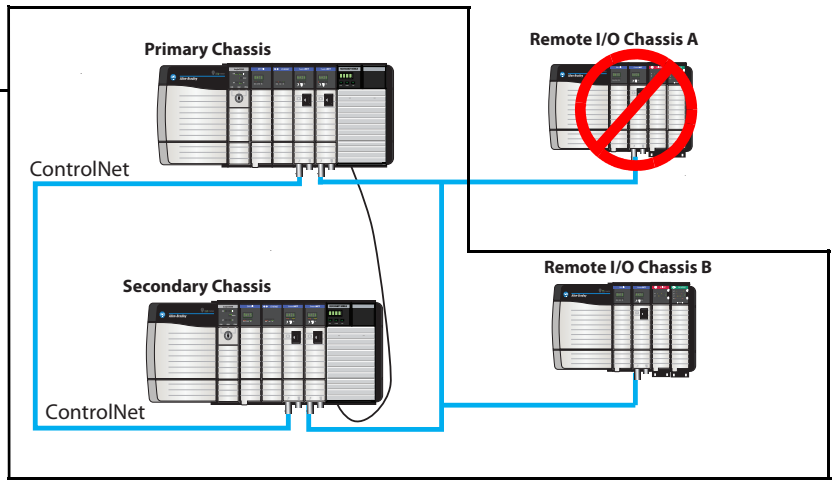
For more information about the specialized I/O termination boards, see Fault-tolerant System Hardware, Chapter 2.

Remote I/O Fault Handling

In the event of a fault in a module or device in one chassis, for example, chassis A, the fault-tolerant system will continue to operate using only the module or device in the other duplicate chassis (chassis B) and the unfaulted modules in chassis A. The system will carry-out the safety function until the faulted module in chassis A is repaired, or until a fault occurs on the corresponding module in chassis B. If a fault in chassis B occurs and chassis A is already faulted the system fails to safe.

Fault Handling with Remote I/O

Despite a fault in chassis A, the rest of the safety system continues to operate.



8 / 2011 www.klinkmann.com KLINKMANN

The Complete ControlLogix Fault-tolerant System

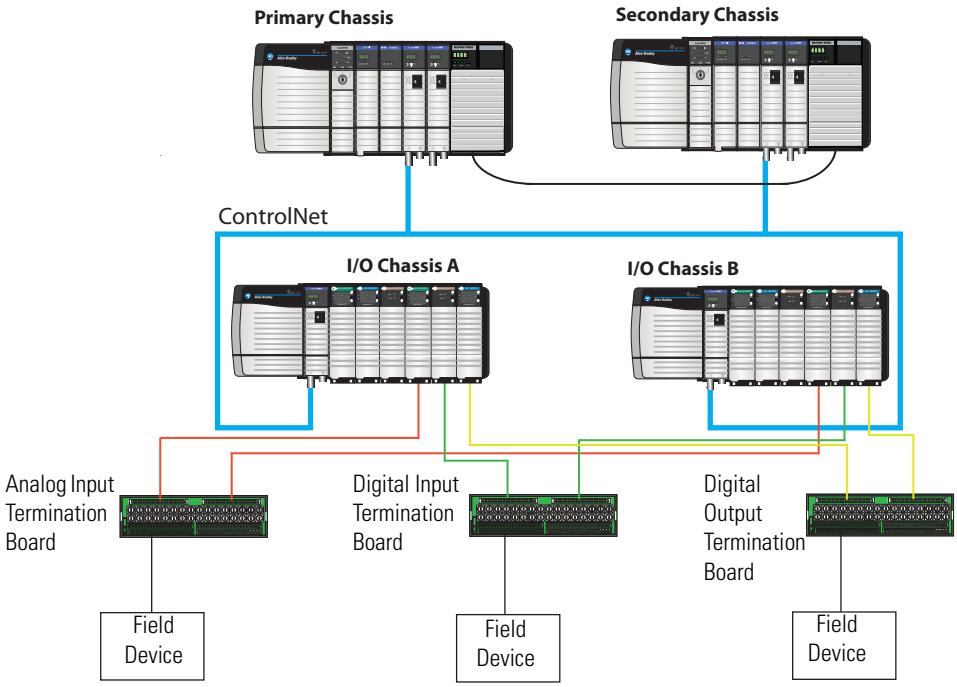
The complete ControlLogix system is comprised of several components that help establish fault tolerance. These components are briefly described here and further described in later chapters.

Hardware

A complete ControlLogix fault-tolerant system, including the redundant controller chassis, duplicate remote I/O chassis, and the specialized termination boards should be configured similar to that shown below.

For more information about the hardware required, see Chapter 2, Fault-tolerant System Hardware, on page 25.

Fault-tolerant Configuration



Software and Programming

The programming and debugging tool required for use with the ControlLogix fault-tolerant system is RSLogix 5000 software, version 15 or later.

Also required are specialized routines developed by Rockwell Automation. The use of these specialized routines are specific only to the fault-tolerant SIL2 configuration.

IMPORTANT A fault-tolerant system configured as described in this manual is SIL2 compliant only when these components are used.

- Hardware specified in Chapter 2.
- RSLogix 5000 software, version 15 or later.
- Routines specific to each type of module pair used.

While the fault-tolerant routines can be used with RSLogix 5000 software, version 15 or later - if you are using RSLogix 5000 software, version 16 or later, you may instead choose to use specialized Add-On Instructions available from Rockwell Automation.

For more information about the SIL2 fault-tolerant Add-On Instructions, see the ControlLogix SIL2 Fault-tolerant Configuration Application Technique manual, publication 1756-AT012. That manual contains information specific to the configuration and use of the SIL2 fault-tolerant Add-On Instructions.

Additional Resources

Resource	Description
ControlLogix Redundancy System User Manual, publication 1756-UM523	This user manual explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.
ControlLogix Fault-tolerant SIL2 Configuration (Using Add-On Instructions) Application Technique, publication 1756-AT012.	The application technique manual describes how to configure and program a fault-tolerant SIL2 system using specialized Add-On Instructions available from Rockwell Automation.
Logix5000 Controllers Add-On Instructions, publication 1756-PM010	This programming manual describes Add-On Instructions and their use in RSLogix 5000 software.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Notes:

Fault-tolerant System Hardware

About This Chapter

This chapter describes the use of the remote I/O and termination boards, including their features and functions, in a ControlLogix fault-tolerant system.

Topic	Page
Approved I/O Modules and Termination Boards	25
About the Specialized Termination Boards	26
1756-IB32 DC Input Termination Board Features	26
Normal Operation of 1756-IB32, DC Input Termination Board	27
1756-IB32 DC Input Termination Board and Transition Tests	28
1756-IF16 Analog Input Termination Board	30
Normal Operation of the 1756-IF16, Analog Input Termination Board	31
1756-IF16 Module Pair Reference Tests	34
1756-OB16D Diagnostic Output Termination Board Features	37
Normal Operation of the 1756-OB16D Diagnostic Output Termination Board	38
Termination Board Relay Control	40
1756-IB32 Input Termination Board Relay Control	40
1756-IF16 Analog Input Termination Board Switch Control	41
1756-OB16D Output Termination Board Relay Control	42
Input Module Diagnostic Test Control	44
Additional Resources	45

Approved I/O Modules and Termination Boards

Only three I/O modules are approved for use in the ControlLogix fault-tolerant system. In addition to the approved I/O modules, specialized termination boards must be used in a fault-tolerant system.

SIL2-approved I/O Modules and Termination Boards

I/O Module Cat. No.	Module Description	Termination Board Cat. No.
1756-IB32	Digital DC Input Module	1492-TIFM40F-F24A-2
1756-IF16[1]	Analog Input Module	1492-TAIFM16-F-3
1756-OB16D	Diagnostic DC Output Module	1492-TIFM40F-24-2

(1) If you are using 1756-IF16 analog input modules in your system, only two-wire transmitters may be used.

8 / 2011 www.klinkmann.com KLINKMANN

About the Specialized Termination Boards

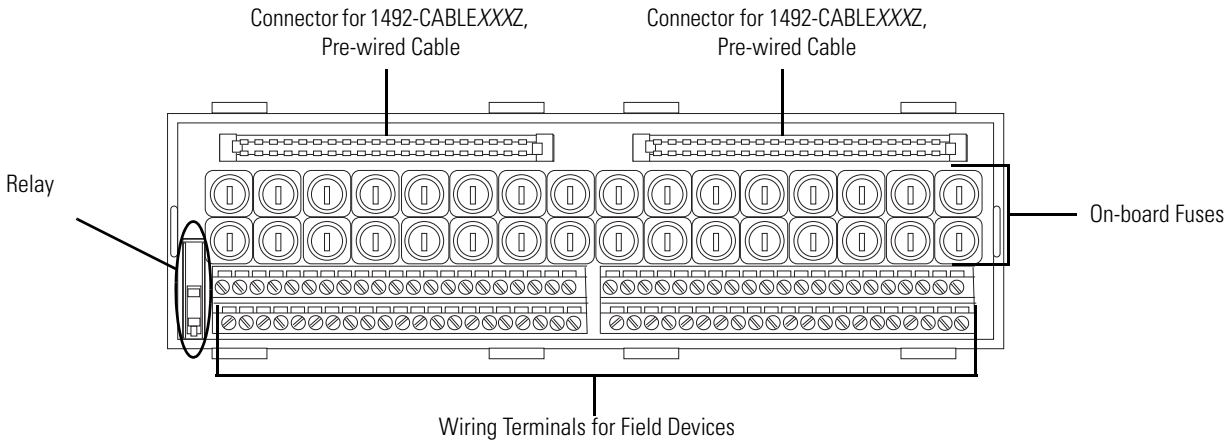
The specialized I/O termination boards (1492-TIFM40F-F24A-2, 1492-TAIFM16-F-3, and 1492-TIFM40F-24-2) are crucial to the implementation of a ControlLogix fault-tolerant system. The functionality of these boards, coupled with the application program developed by Rockwell Automation, make fault-tolerant I/O configurations possible.

1756-IB32 DC Input Termination Board Features

The specialized digital input termination boards, catalog number 1492-TIFM40F-F24A-2, have these hardware features:

- On-board fusing with status indicators
- Easy-to-use wiring terminals
- Relay for diagnostic tests
- Pre-wired cables for use from termination board to I/O module

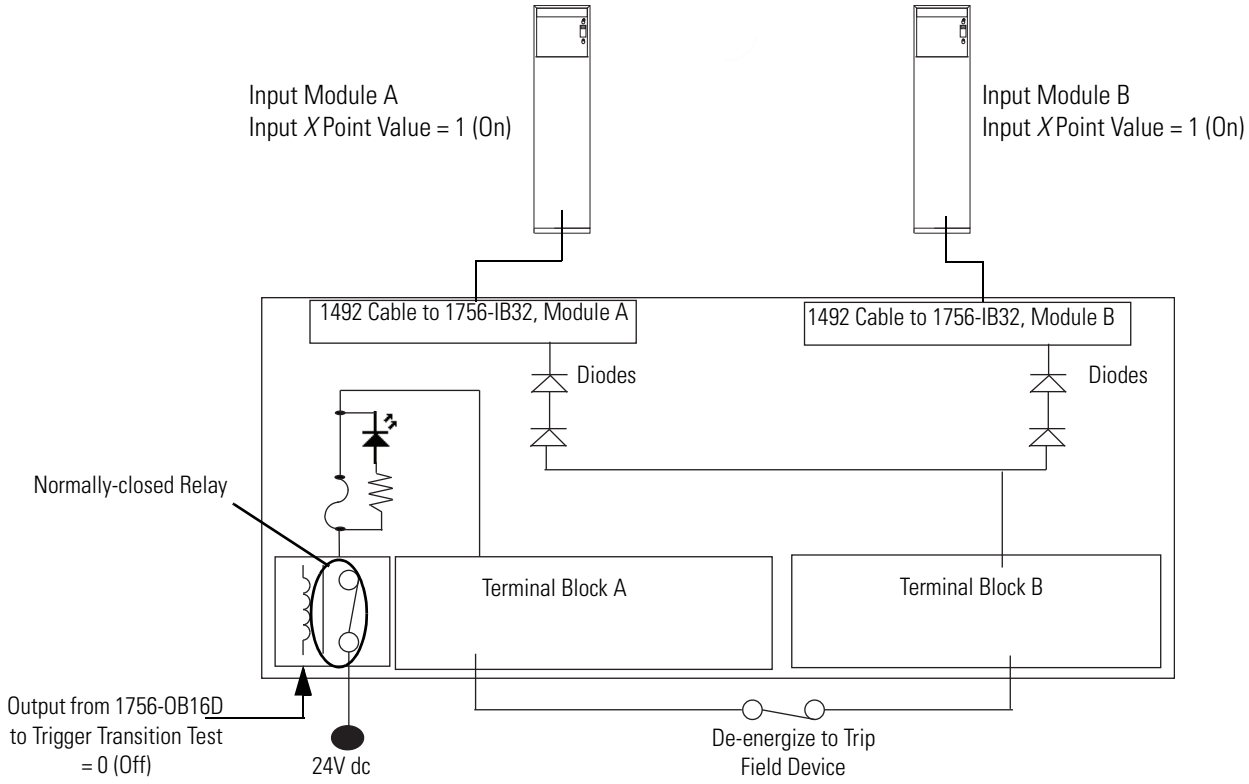
DC Input Termination Board for Use with 1756-IB32 Input Modules



Normal Operation of 1756-IB32, DC Input Termination Board

During normal operation, the digital input termination board functions as shown in the diagram below.

1492-TIFM40F-F24A-2, Digital Input Termination Board - Normal Operation



Note that this graphic represents only one of several possible field device inputs.

During normal operation (that is, when a diagnostic test is not in progress), the primary function of the termination board is to route one de-energize-to-trip sensor to the same two duplicate input points, one on each module of the 1756-IB32 pair.

As shown in the diagram above, 24V dc field power is routed through the normally-closed relay. It then passes through a fuse and to the sensors connected to wiring terminals A and B.

The on/off status is then routed through the isolating diodes, and through the cables that connect the termination board to the input modules.

1756-IB32 DC Input Termination Board and Transition Tests

In the fault-tolerant system, diagnostic tests are carried-out on the 1756-IB32 module pair. These diagnostic tests are called transition tests. The transition tests verify that the input points of the 1756-IB32 module pair are able to transition from on to off when required.

Transition Test Intervals

Transition tests are programmed in the specialized program supplied by Rockwell Automation. They occur at a user-specified intervals based upon the requirements of the SIL2 application.

If there are no faults present on the 1756-IB32 module pair, the system operates using the test interval specified in the tag ModulePair_Good_TestInterval. If the system is operating using only data from one module of the pair (that is, in a 1oo1 state) the transition tests occur more frequently as specified in the tag ModulePair_1oo1_TestInterval.

This table shows the test interval tags and the recommended interval values.

Transition Test Interval Tags

Tag Name	Recommended Value
ModulePair_Good_TestInterval	86,400,000 (24 hours)
ModulePair_1oo1_TestInterval	3,600,000 (1 hour)

Termination Board During Transition Tests

During the transition test, an output from a diagnostic output module pair[(1)] triggers the normally-closed relay of the 1756-IB32 input termination board to open. Thus, power is temporarily removed from the field sensors.

Each point is checked for an off status. If the point did not transition to off, then that point is identified by the program as stuck-at-one and is processed as a fault. If the points transition successfully, then the normally-closed relay is switched from open to closed, re-applying power to the sensors.

(1) To achieve fault tolerance, diagnostic tests for the input module pair should be triggered only by outputs from the 1756-OB16D module pair. In addition, 1756-OB16D module outputs that are being used to trigger the diagnostic tests should have pulse tests disabled. For more information about disabling pulse tests for outputs, see Edit ModulePair Tags on page 76.

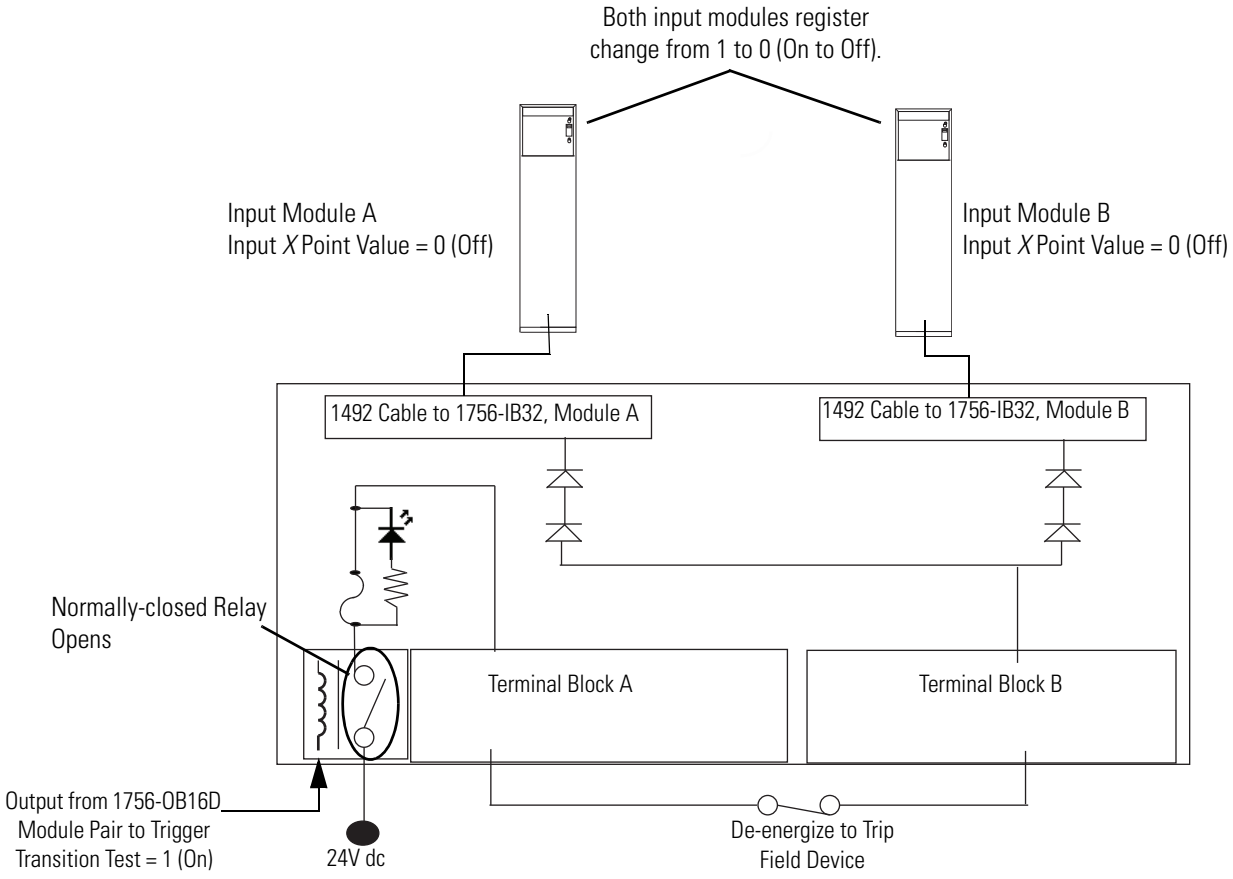
While this transition occurs, the specialized program continues to control the system based upon the last-known and verified data from the modules.

IMPORTANT The transition test detects only stuck-at-one conditions.

Any zero (or low) condition on any point of the module pair is recognized by the controller as a demand on the safety system.

This graphic depicts the function of the input termination board during a transition test.

Digital Input Module Termination Board Functions During Transition Test


Both input modules register
change from 1 to 0 (On to Off).
Input Module A
Input X Point Value = 0 (Off)
Input Module B
Input X Point Value = 0 (Off)
1492 Cable to 1756-IB32, Module A
1492 Cable to 1756-IB32, Module B
Normally-closed Relay
Opens
Terminal Block A
Terminal Block B
Output from 1756-OB16D
Module Pair to Trigger
Transition Test = 1 (On)
24V dc
De-energize to Trip
Field Device

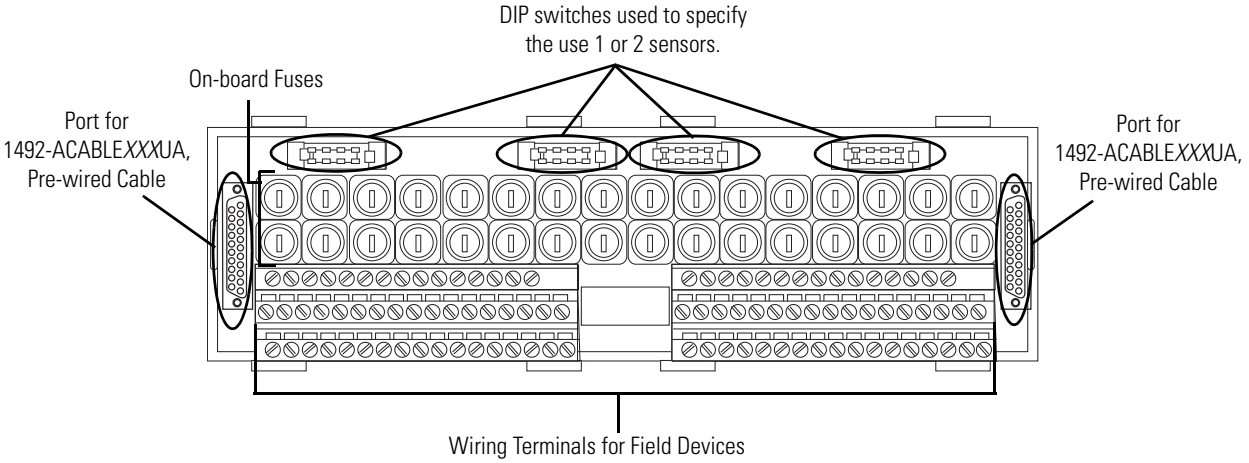
Note that this graphic represents only one of several possible field device inputs.

1756-IF16 Analog Input Termination Board

The specialized analog input termination boards have these hardware features:

- On-board fusing with status indicators
- Easy-to-use wiring terminals
- On-board reference voltages and solid-state switches for diagnostic tests
- Pre-wired cables for use from termination board to I/O module
- DIP switch selection for easy use of one or two-sensor wiring

Analog Input Termination Board for Use with 1756-IF16 Input Modules


DIP switches used to specify the use 1 or 2 sensors.
On-board Fuses
Port for 1492-ACABLE*XXX*UA, Pre-wired Cable
Port for 1492-ACABLE*XXX*UA, Pre-wired Cable
Wiring Terminals for Field Devices

Normal Operation of the 1756-IF16, Analog Input Termination Board

During normal operation (that is, when a diagnostic test is not in progress), the primary purpose of the analog termination board is to route 2-wire transmitters to input channels, one on each module of the pair.

The analog termination board provides the capability to wire one or two sensors to each input channel.

For more information about one- and two-sensor wiring, see the section titled One-sensor or Two-sensor Wiring Option on page 33.

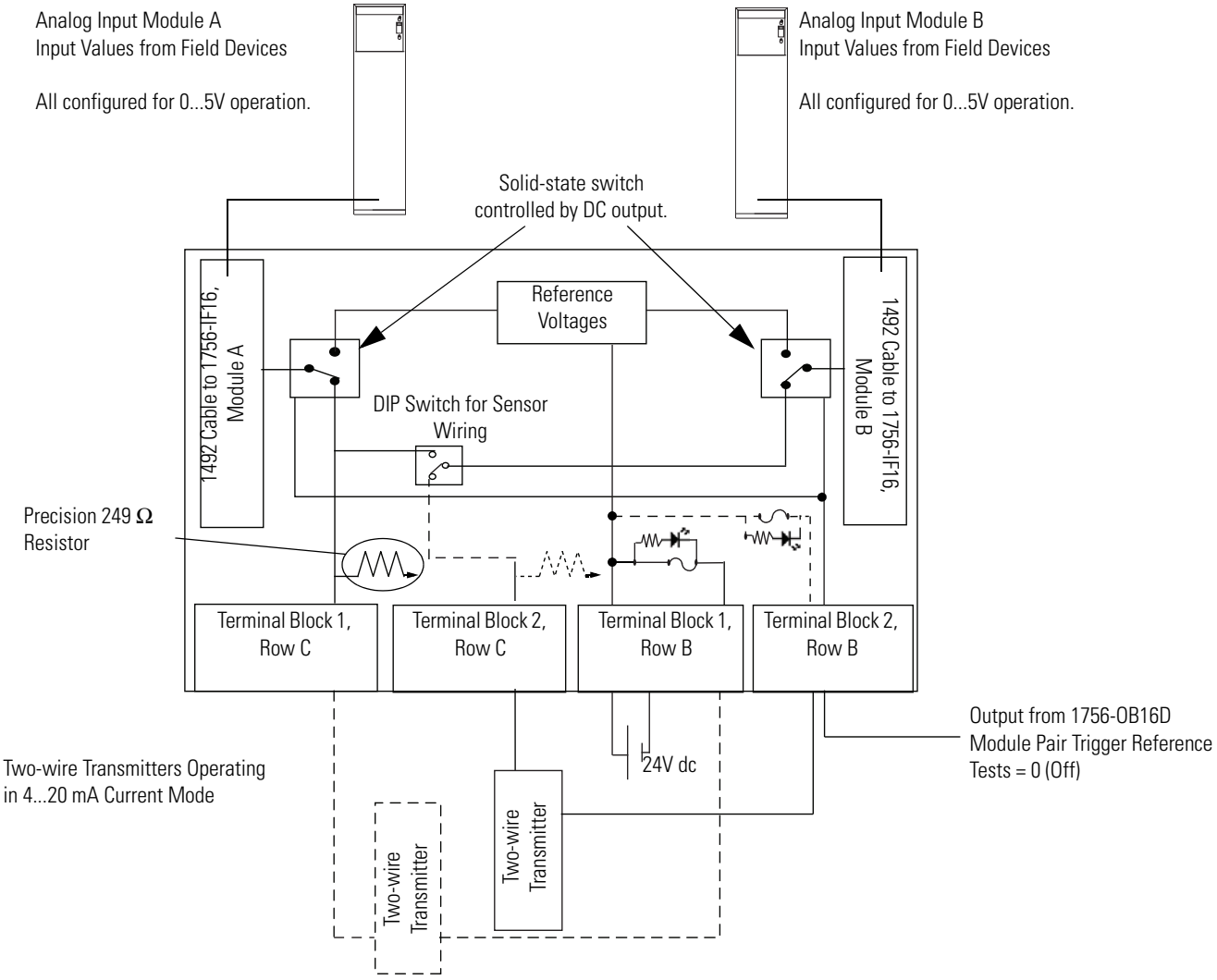
Two-wire transmitters operate in 4...20 mA current mode powered by 24V dc. The 4...20 mA signals are converted to voltage by the on-board precision 249 Ω resistor. The voltage is then routed to the same two duplicate input channels, one on each module of the 1756-IF16 pair. Each 1756-IF16 module is configured for 0…5V operation.

The application program supplied by Rockwell Automation then compares the two channel values to each other and verifies that the values are within the user-defined deadband value. The two channels' values are then averaged and made available for use by the program.

8 / 2011 www.klinkmann.com KLINKMANN

During normal operation, the analog input termination board functions as depicted in this diagram.

1492-TAIFM16-F-3, Analog Input Termination Board - Normal Operation


Analog Input Module A
Input Values from Field Devices
All configured for 0...5V operation.
Analog Input Module B
Input Values from Field Devices
All configured for 0...5V operation.
Solid-state switch
controlled by DC output.
1492 Cable to 1756-IF16, Module A
Reference Voltages
DIP Switch for Sensor Wiring
1492 Cable to 1756-IF16, Module B
Precision 249 Ω Resistor
Terminal Block 1, Row C
Terminal Block 2, Row C
Terminal Block 1, Row B
Terminal Block 2, Row B
Output from 1756-OB16D
Module Pair Trigger Reference
Tests = 0 (Off)
24V dc
Two-wire Transmitters Operating in 4...20 mA Current Mode
Two-wire Transmitter
Two-wire Transmitter

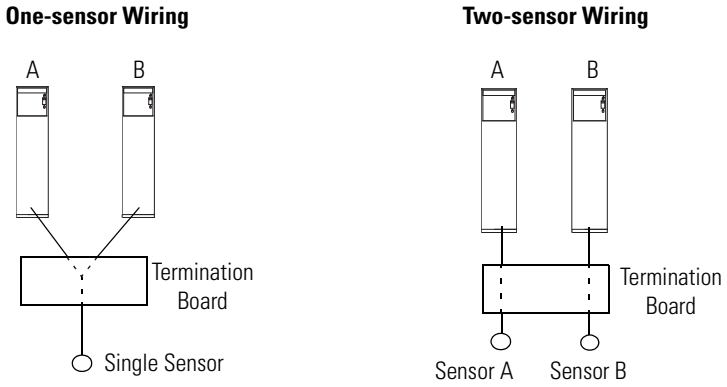
– – – – Dashed line represents the preferred method of wiring, that is, the use of two-sensor wiring.

Note that this graphic represents only one of several possible field device inputs.

One-sensor or Two-sensor Wiring Option

The DIP switches located at the top of the analog input termination board are used to specify one- or two-sensor wiring. One-sensor wiring should be used when one field-sensor signal is being routed to the same channel on to two separate input modules of the pair. Two-sensor wiring should be used when two-sensor signals are routed through the board to the same two separate channels, one on each module of the pair.

One- and Two- Sensor Wiring



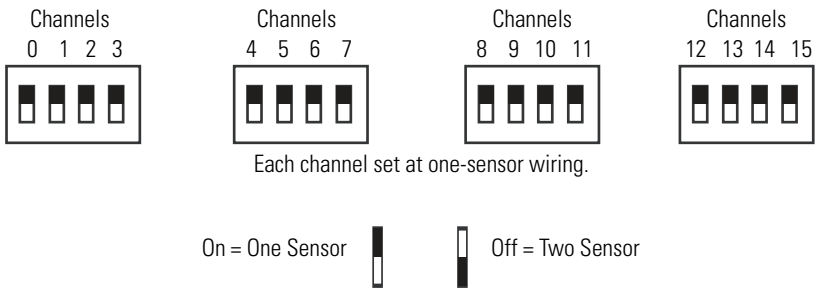
The default of DIP switches on the termination board is to one-sensor wiring. You may choose to use a combination of one- and two-sensor wiring on the analog termination board.

IMPORTANT If you use one-sensor wiring, you must configure the 1756-IF16 module pair reference tests to occur more frequently than the safety response time of your application.

For information about configuring the reference tests, see the section Recommended 1756-IF16 ModulePair Tag Values, on page 80.

Use the diagrams below as a reference when using the DIP switch to set one- or two-sensor wiring.

1492-TAIFM16-F-3, Analog Input Termination Board DIP Switch Designations


Channels 0 1 2 3
Channels 4 5 6 7
Channels 8 9 10 11
Channels 12 13 14 15
Each channel set at one-sensor wiring.
On = One Sensor
Off = Two Sensor

8 / 2011 www.klinkmann.com KLINKMANN

1756-IF16 Module Pair Reference Tests

The 1756-IF16 diagnostic tests are called reference tests. The results of the reference tests are used by the application program to verify that the analog modules are capable of accurately reading analog data values. While the test is carried-out by the termination board, the control program continues to run on last-known data (that is, the most recent data validated by the program).

Reference Test Intervals

Reference tests are programmed in the specialized program supplied by Rockwell Automation. They occur at a user-specified intervals based upon the requirements of the SIL2 application.

If there are no faults present on the 1756-IF16 module pair, the system operates using the test interval specified in the tag ModulePair_Good_TestInterval. If the system is operating using only data from one module of the pair (that is, in a 1oo1 state) the reference tests occur more frequently as specified in the tag ModulePair_1oo1_TestInterval.

Reference test intervals are specified in these ModulePair tags.

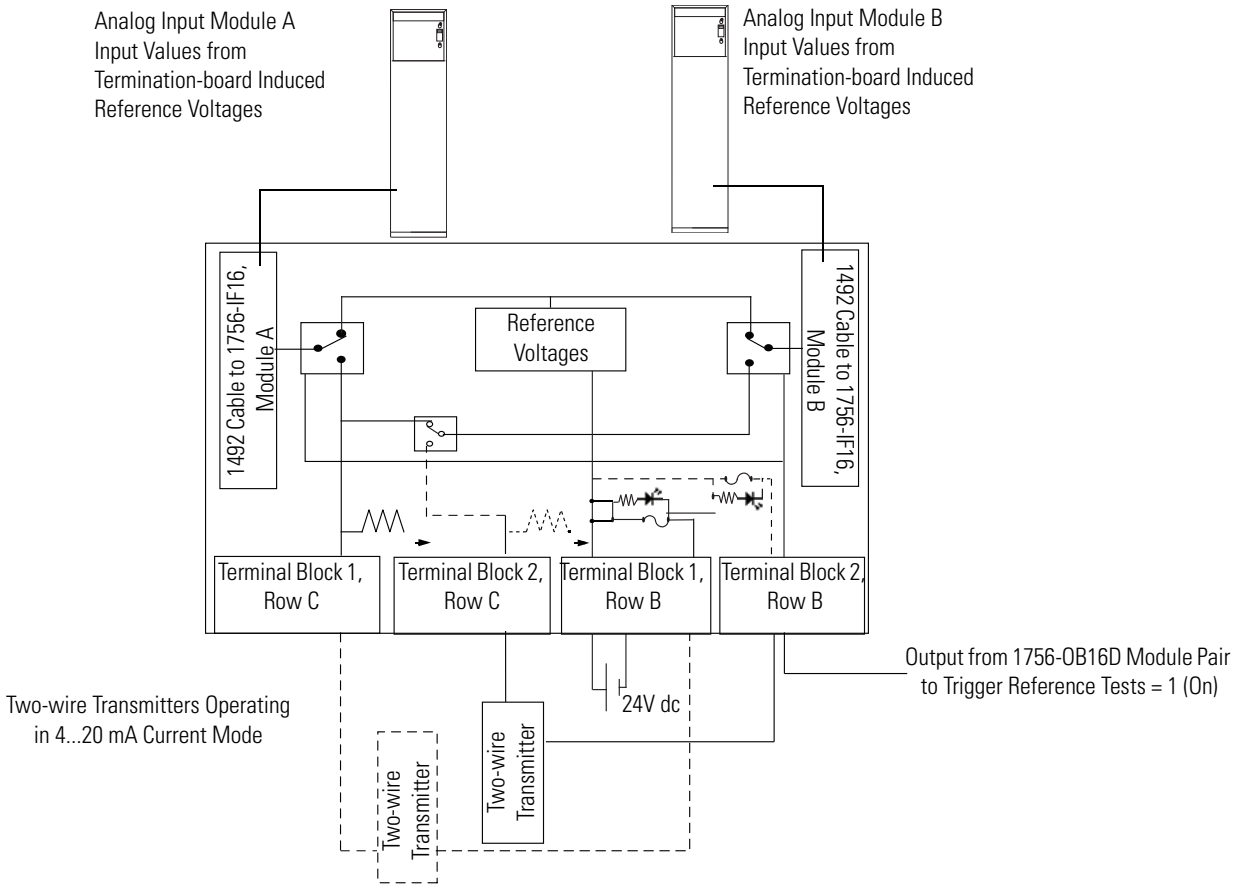
Reference Test Tags

Tag Name	Recommended Value
ModulePair_Good_TestInterval	86,400,000 (24 hours)
ModulePair_1oo1_TestInterval	3,600,000 (1 hour)

Termination Board During Reference Tests

When a reference test is initiated, the analog termination board functions as depicted below.

1492-TAIFM16-F-3, Analog Input Termination Board During Reference Test



– – – – – Dashed line represents the preferred method of wiring, that is, the use of two-sensor wiring.

Note that this graphic represents only one of several possible field device inputs.

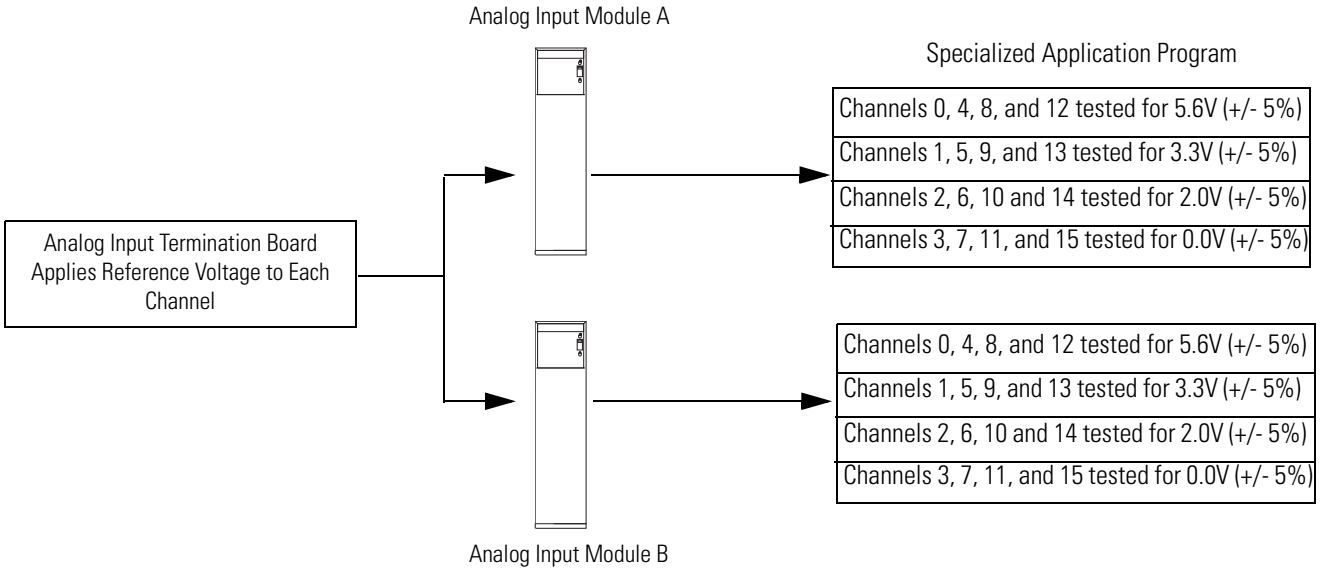
As depicted, the output from the 1756-OB16D module pair triggers(1) the analog input termination board to switch from the field device voltages to the reference voltages. Each channel has a specific reference voltage applied. This table shows each channel and corresponding reference voltage.

1756-IF16 Reference Voltages

Channel No.	Reference Voltage
0, 4, 8, and 12	5.6V
1, 5, 9, and 13	3.3V
2, 6, 10, and 14	2.0V
3, 7, 11, and 15	0.0V

The program verifies that the 1756-IF16, analog input channels correctly read the reference values within +/- 5% (the default value as specified in the `ReferenceTest_Deadband[X]` tag.

Analog Input Module Reference Test


Analog Input Module A
Specialized Application Program
Channels 0, 4, 8, and 12 tested for 5.6V (+/- 5%)
Channels 1, 5, 9, and 13 tested for 3.3V (+/- 5%)
Channels 2, 6, 10 and 14 tested for 2.0V (+/- 5%)
Channels 3, 7, 11, and 15 tested for 0.0V (+/- 5%)
Analog Input Termination Board Applies Reference Voltage to Each Channel
Channels 0, 4, 8, and 12 tested for 5.6V (+/- 5%)
Channels 1, 5, 9, and 13 tested for 3.3V (+/- 5%)
Channels 2, 6, 10 and 14 tested for 2.0V (+/- 5%)
Channels 3, 7, 11, and 15 tested for 0.0V (+/- 5%)
Analog Input Module B

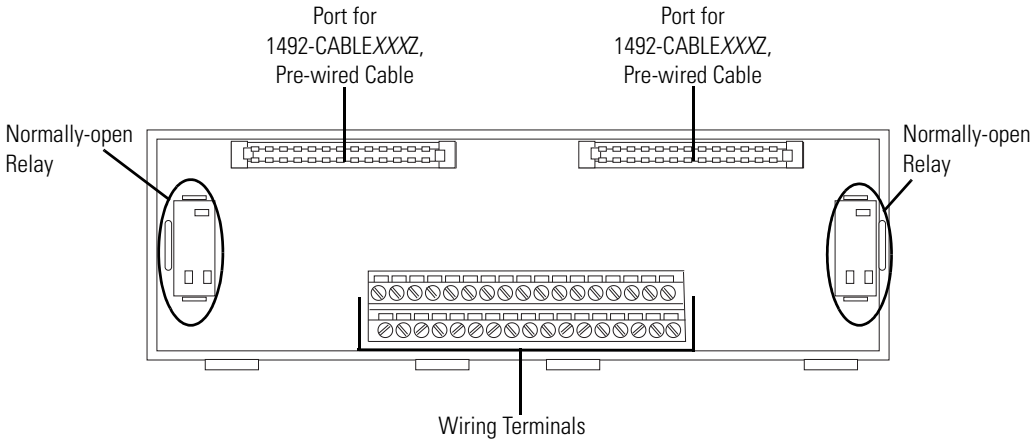
(1) To achieve fault-tolerance, diagnostic tests for the input module pair should be triggered only by outputs from the 1756-OB16D module pair. In addition, 1756-OB16D module outputs that are being used to trigger the diagnostic tests should have pulse tests disabled. For more information about disabling pulse tests for outputs, see Edit ModulePair Tags on page 76.

1756-OB16D Diagnostic Output Termination Board Features

The specialized output termination boards have these hardware features:

- Easy-to-use wiring terminals
- Relays to provide secondary method of power disconnect for each output module connected
- Pre-wired cables for use from termination board to I/O module
- On-board blocking diodes isolate output points

Diagnostic Output Termination Board for Use with 1756-OB16D Input Modules



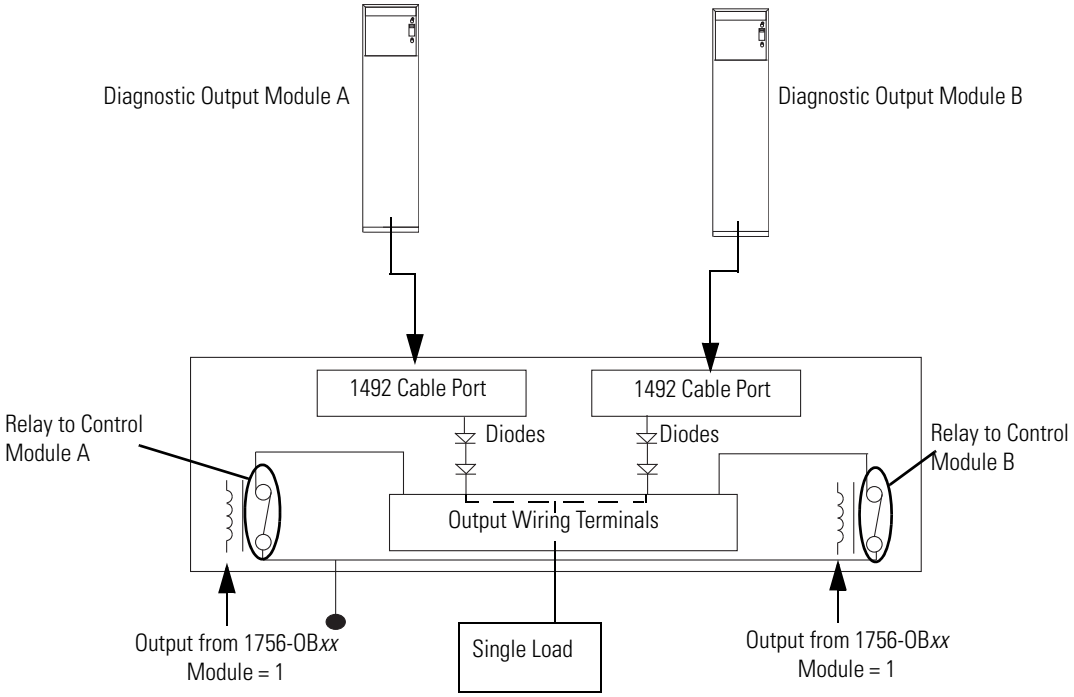
8 / 2011 www.klinkmann.com KLINKMANN

Normal Operation of the 1756-OB16D Diagnostic Output Termination Board

During normal operation, the primary function of the 1756-OB16D, output termination board is to connect the same two output points, each from one module of the pair, to a single load. The output termination board also provides isolation for each channel through the use of diodes.

A normally-open relay is held closed by a nonfault-tolerant, DC output from the system. While the relay is closed, power to each 1756-OB16D module of the pair is provided.

Diagnostic Output Termination Board Functions



Diagnostic Tests and the 1756-OB16D Output Termination Board

Because the 1756-OB16D modules have on-board diagnostic features, the only interaction between the output termination board and diagnostic tests occurs if a module fails a diagnostic test.

If the diagnostic tests find a module fault, power is disconnected from the faulted module by opening the normally-open relay on the output termination board. The disconnect is triggered by an output of a designated 1756-OB*xx* module.

For more information about the 1756-OB*xx* modules and disconnects, see the section titled 1756-IF16 Analog Input Termination Board Switch Control on page 41.

8 / 2011 www.klinkmann.com KLINKMANN

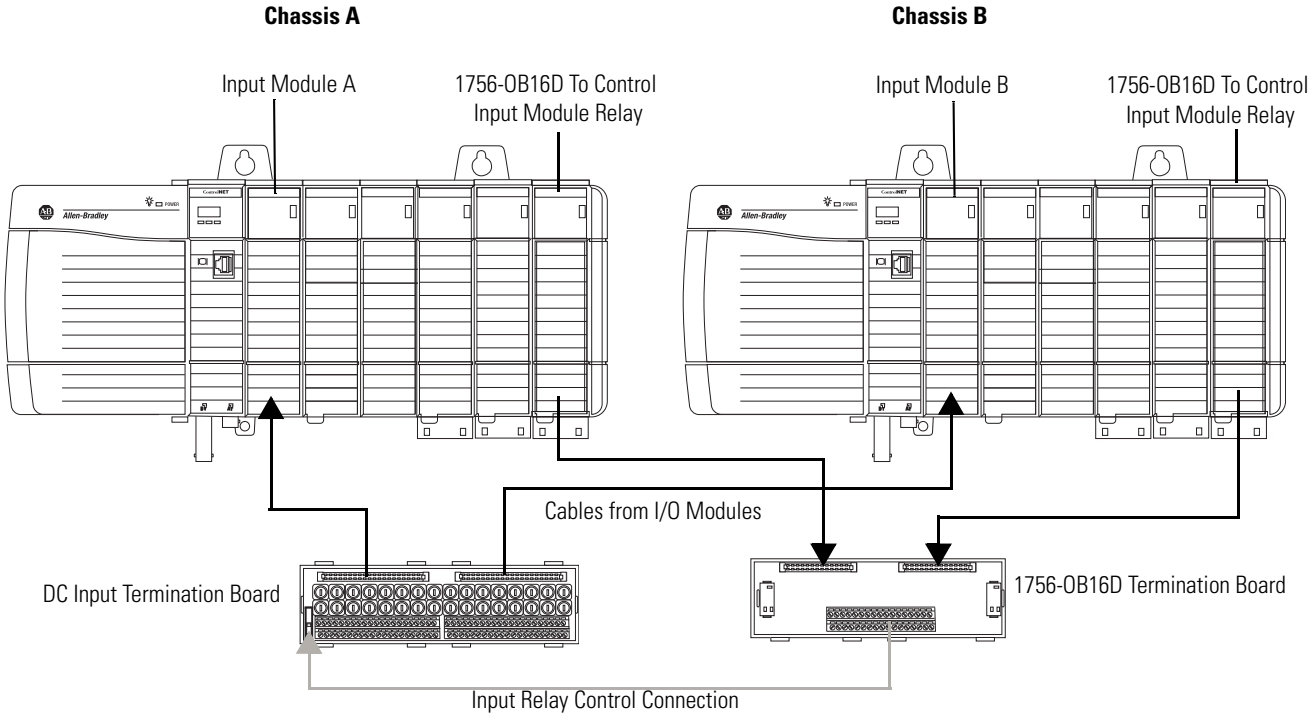
Termination Board Relay Control

Both the input module pairs and the output module pairs require the use of output points to control some actions of the termination boards. Each type of module pair (input and output) has different requirements for termination board relay control.

1756-IB32 Input Termination Board Relay Control

In order to establish high availability for the execution of transition tests, the relay on the DC input termination boards is controlled by an output from the 1756-OB16D module pair. The signal from this output is used to initiate transition tests.

DC Input Termination Board Relay Control

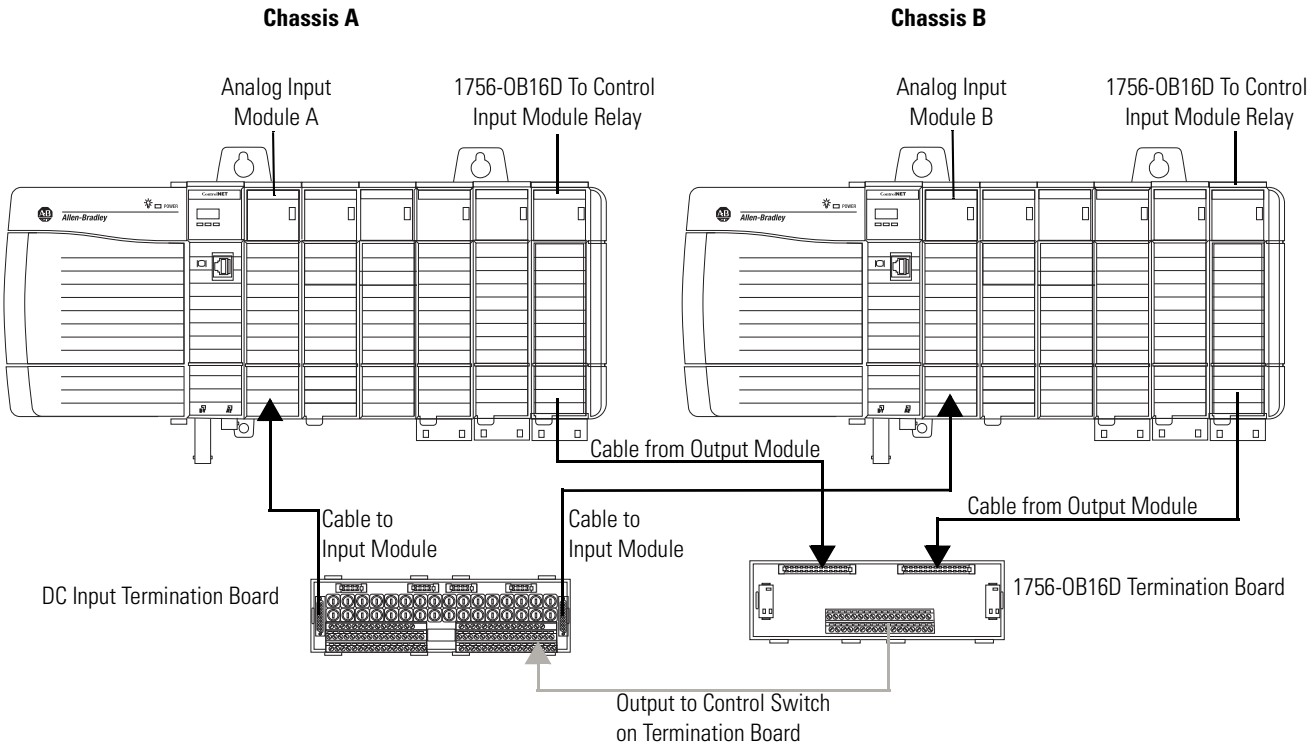


IMPORTANT You must disable pulse tests on outputs of the 1756-OB16D module pair that are connected to input termination boards.

1756-IF16 Analog Input Termination Board Switch Control

In order to establish high availability for the execution of reference tests, the switch on the analog input termination boards is controlled by an output from the 1756-OB16D module pair. The signal from this output is used to initiate reference tests.

Analog Input Termination Board Relay Control


Chassis A
Chassis B
Analog Input Module A
1756-OB16D To Control Input Module Relay
Analog Input Module B
1756-OB16D To Control Input Module Relay
Cable from Output Module
Cable to Input Module
Cable to Input Module
Cable from Output Module
DC Input Termination Board
1756-OB16D Termination Board
Output to Control Switch on Termination Board

IMPORTANT You must disable pulse tests on outputs of the 1756-OB16D module pair that are connected to input termination boards.

8 / 2011 www.klinkmann.com KLINKMANN

1756-OB16D Output Termination Board Relay Control

To control relays on the 1756-OB16D termination board, use at least two SIL2-certified output modules. The SIL2-certified modules available for use are listed here.

- 1756-OB16I
- 1756-OB8EI
- 1756-OB32
- 1756-OB16D

IMPORTANT The 1756-OB*xx* modules must be placed in the same chassis as the 1756-OB16D module whose relay it is controlling.

For example, a 1756-OB*xx* module in chassis A should be placed and connected to control the relay of a 1756-OB16D (one of the module pair) module in chassis A.

Use of 1756-OB16D Modules for Relay Control

If you use two 1756-OB16D modules to control the relays of an output termination board, make these considerations.

IMPORTANT Do not use the two 1756-OB16D modules used to control the output relays as a module pair.

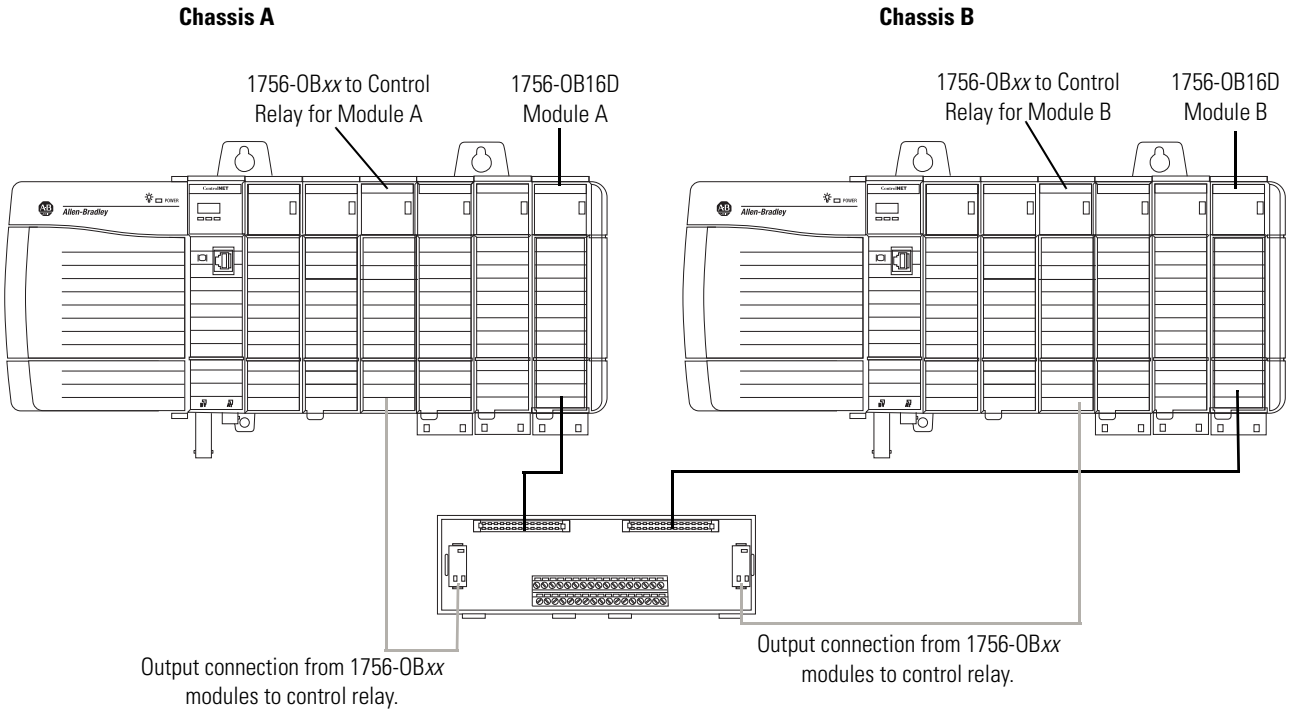
IMPORTANT If you use 1756-OB16D modules to control the output termination board relays, you must disable pulse testing for those output points.

Failing to disable pulse testing on output points designated to control termination board relays may result in unintended and potentially hazardous disconnects.

Because you must use the 1756-OB*xx* module in the same chassis as the 1756-OB16D module whose relay it is controlling, you may want to group all of your 1756-OB16D modules in designated output chassis pairs. Doing so will reduce the number of 1756-OB*xx* you must use to control output relays.

See Appendix on page 149 for more information.

1756-OB*xx* Modules to Control 1756-OB16D Termination Board Relays


Chassis A
Chassis B
1756-OBxx to Control Relay for Module A
1756-OB16D Module A
1756-OBxx to Control Relay for Module B
1756-OB16D Module B
Output connection from 1756-OBxx modules to control relay.
Output connection from 1756-OBxx modules to control relay.

For more information about SIL2-certified output modules, see Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001.

Input Module Diagnostic Test Control

Control of the input diagnostic tests (that is, the transition and reference tests) is achieved through the use of 1756-OB16D outputs routed through the 1756-OB16D termination board.

Because the 1756-OB16D outputs are used to control the diagnostic tests, any fault that results in the shutdown of the 1756-OB16D module pair will result in the failure of the next transition or reference tests for the input modules. This is due to the inability of the disconnected outputs to initiate the diagnostic tests.

For more information about the control of input diagnostic tests, see these sections:

- 1756-IB32 Input Termination Board Relay Control, page 40
- 1756-IF16 Analog Input Termination Board Switch Control, page 41

Hardware and Programming

In order to achieve fault tolerance, you must use the hardware described in this chapter as well as the program supplied by Rockwell Automation. The program, its elements, and configuration are described in the chapters titled Fault-tolerant Program Elements (on page 25) and Configuring the Fault-tolerant System (on page 65).

Additional Resources

Resource	Description
1756-IB32 Termination Board Installation Instructions, publication 41063-290-01	Provides a description of installation procedures and a wiring diagram for the 1756-IB32 termination board.
1756-IF16 Termination Board Installation Instructions, publication 41063-292-01	Provides a description of installation procedures and a wiring diagram for the 1756-IF16 termination board.
1756-OB16D Termination Board Installation Instructions, publication 41063-291-01	Provides a description of installation procedures and a wiring diagram for the 1756-OB16D termination board.
ControlLogix 32-Point DC (10-31.2V) Input Module Series B Installation Instructions, publication 1756-IN027	Provides installation procedures and a wiring diagram for 1756-IB32, digital input module.
ControlLogix Voltage/Current Input Module Installation Instructions, publication 1756-IN039	Provides installation procedures and a wiring diagram for 1756-IF16, analog input module.
ControlLogix DC (19.2-30V) Diagnostic Output Module Installation Instructions, publication 1756-IN058	Provides installation procedures and a wiring diagram for 1756-OB16D, diagnostic output module.
ControlLogix Chassis, Series B Installation Instructions, publication 1756-IN080	Provides installation procedures for ControlLogix chassis.
ControlLogix 32-Point DC (10-31.2V) Input Module Series B Install. Instructions, publication 1756-IN027	Provides wiring diagrams, step-by-step installation instructions, and module specifications.
Bul 1492 Fused Term. Module for use in SIL2 Safety Shutdown Appl. w/2 1756-IB32, publication 41603-290-01	Provides wiring schematics and installation instructions for the termination board.
ControlLogix Voltage/Current Input Module Installation Instructions, publication 1756-IN039	Provides wiring diagrams, step-by-step installation instructions, and module specifications.
Bul 1492 Fused Term. Module for use in SIL2 Safety Shutdown Appl. w/2 1756-IF16D, publication 41063-292-01	Provides wiring schematics and installation instructions for the termination board.
ControlLogix DC (19.2-30V) Diagnostic Output Module, publication 1756-IN058	Provides wiring diagrams, step-by-step installation instructions, and module specifications.
Bul 1492 Fused Term. Module for use in SIL2 Safety Shutdown Appl. w/2 1756-OB16D, publication 41063-291-01	Provides wiring schematics and installation instructions for the termination board.
ControlLogix Digital I/O Modules User Manual, publication 1756-UM058	Provides information about digital I/O modules including: features, configuration, and troubleshooting.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Fault-tolerant Program Elements

About This Chapter

This chapter describes some of the elements of the fault-tolerant program provided by Rockwell Automation. The concepts of this chapter should be understood before you configure your system.

Topic	Page
Overview of the Program Elements	47
Main Routine	47
Diagnostic Subroutines	48
Call_Code Subroutines	49
Function of the Program Elements	50
Program Elements Provided	51
States of the System	52
IB32_Diagnostics Subroutine	55
IF16_Diagnostics Subroutine	57
IF16_RefCal Subroutine	59
OB16D_Diagnostics Subroutine	60
Data Flow Between Program Elements	62
Additional Resources	63

Overview of the Program Elements

The following sections provide an overview of the main elements used in the programming for a SIL2-certified, fault-tolerant system.

Main Routine

The main routine of the program is user-programmed based on the requirements for the SIL2 system being implemented. It uses data processed and outputted by the diagnostic subroutines to determine system behavior.

For more information about programming the main routine, see Chapter 5, Programming the Fault-tolerant System, on page 47.

Diagnostic Subroutines

The program supplied by Rockwell Automation contains diagnostic subroutines that must be used to monitor, process, and reconcile data from the input and output module pairs. The data that the subroutines produce is used in the main routine.

Fully-programmed diagnostic subroutines are provided in the program and must be run for each module pair in system. For each type of I/O module certified for use in the SIL2 fault-tolerant system, a diagnostic subroutine is provided.

Module-specific Diagnostic Subroutines

Module Cat. No.	Diagnostic Subroutine Name
1756-IB32	IB32_Diagnostics
1756-IF16	IF16_Diagnostics
1756-OB16D	OB32_Diagnostics

These subroutines are visible in the configuration tree, however, because these diagnostic subroutines are protected, you cannot access or alter them.

Diagnostic Features of Subroutines

The specialized application programming developed by Rockwell Automation executes all of the diagnostic checks and tests described in Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001. Additionally, the specialized application programming executes tests that are specific only to the fault-tolerant configuration.

This table lists the diagnostic features and tests used in a SIL2 system as well as where a description of the feature or test can be found.

Diagnostic Features of Diagnostic Subroutines

For the feature or test	See the description at
Module-level fault reporting	Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001
Data echo communication check	Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001
Field-side output verification	Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001
Pulse testing in the diagnostic output module	Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001

Diagnostic Features of Diagnostic Subroutines

For the feature or test	See the description at
Input comparison	IB32_Diagnostics Subroutine on page 55 and IF16_Diagnostics Subroutine on page 57
Connection verification	Tag descriptions at Appendix A on page 131
Transition tests	1756-IB32 DC Input Termination Board and Transition Tests on page 28
Reference tests	1756-IF16 Module Pair Reference Tests on page 34

Call_Code Subroutines

Each module pair Call_Code subroutine contains:

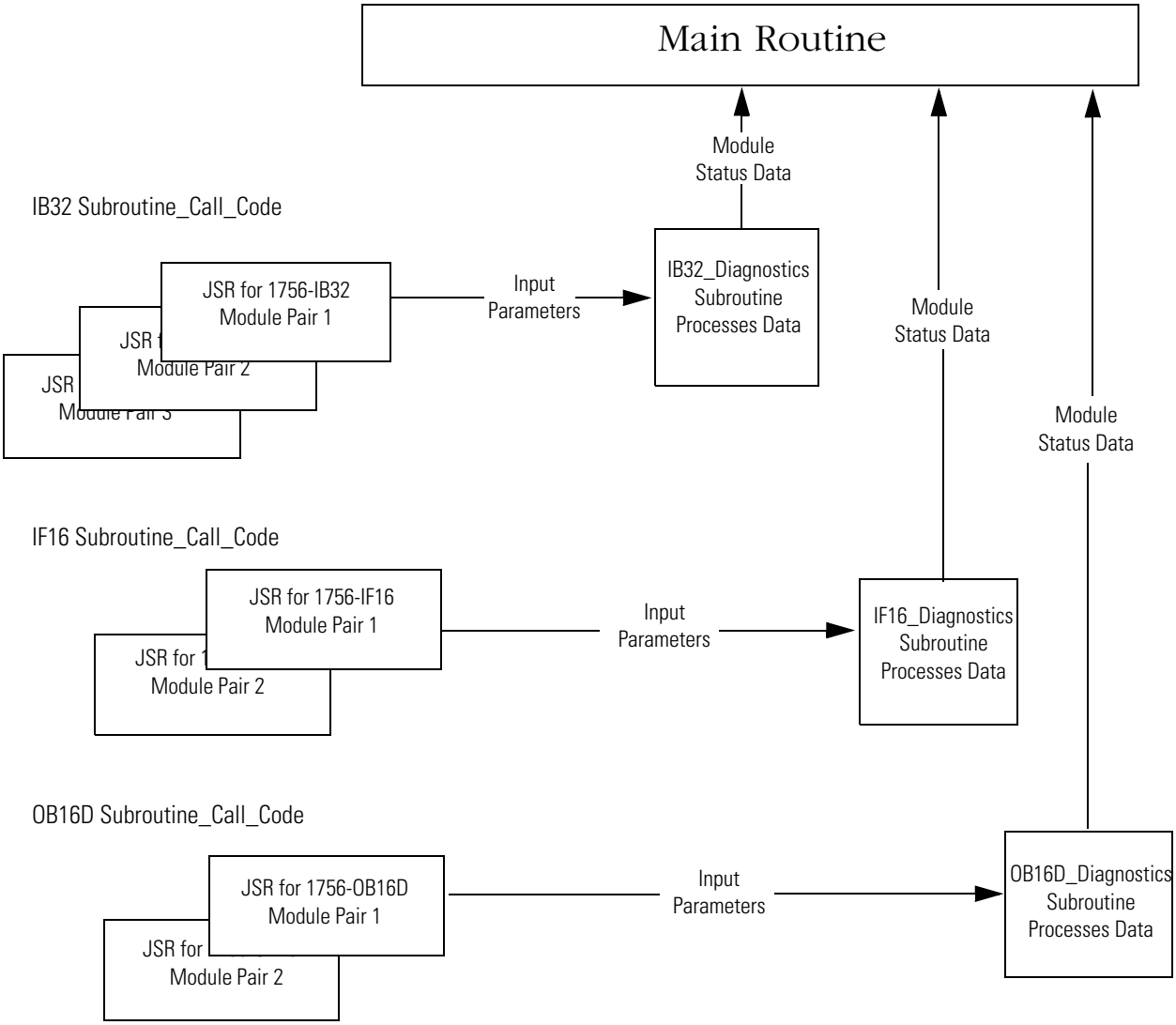
- a JSR instruction that sends and receives data to the diagnostic subroutine for each module pair.
- other programming that initiates diagnostic tests (that is transition and reference tests) for the module pair.

8 / 2011 www.klinkmann.com KLINKMANN

Function of the Program Elements

When configured and programmed properly, the program elements function as depicted here.

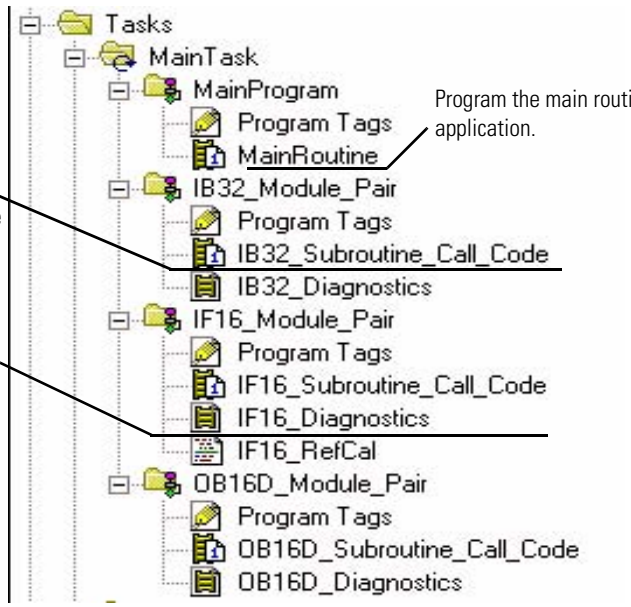
Overview of Fault-Tolerant Program



Program Elements Provided

The fault-tolerant program you receive from Rockwell Automation provides all of the elements described above. The following graphic shows how these elements will appear in the RSLogix 5000 configuration tree.

Program Elements in RSLogix 5000 Configuration Tree



8 / 2011 www.klinkmann.com KLINKMANN

States of the System

To understand how the system diagnostics function, you should understand various states of the system as described in these sections:

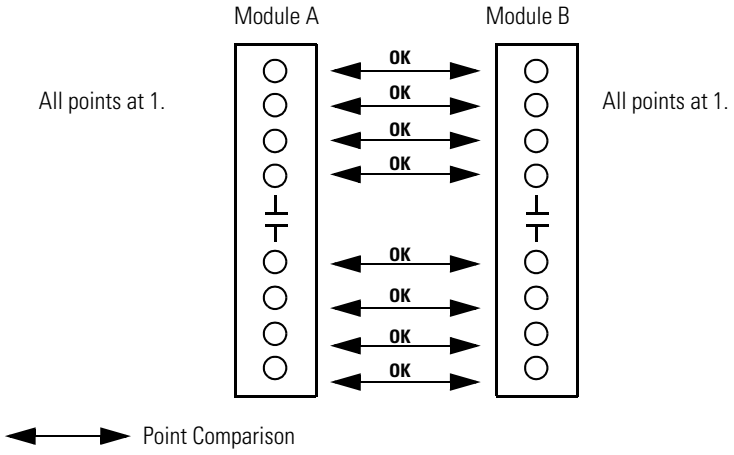
- Normal State see page 52
- Test State see page 52
- 1oo1 State see page 53
- Faulted State see page 54

Normal State

During the normal state:

- no transition or reference test is being carried-out.
- no faults exist in the module pair.
- no demand on the system is present.

Normal Operation - Diagram



Test State

The test state is specific only to the 1756-IB32 and 1756-IF16 modules. During the test state:

- a transition or reference test is being carried-out.
- the system runs on input data from just before the test began.
- no demand on the system is present.

A demand made through the module pair being tested is not processed by the SIL2 system until the test is complete. This is because the system operates on input data from just before the diagnostic test while the diagnostic test is carried out.

For more information about transition and reference tests, see Chapter 2, page 28 and page 34.

1oo1 State

The state when either:

- **A point-level or channel-level fault is present** on one module of the pair. During this state, one or more points of one module of the pair are faulted. The system operates by using data from the unfaulted module *and* all of the unfaulted points of the module with a fault.

  The diagram titled 1oo1 Due to a Point or Channel Fault (below) illustrates this concept.

IMPORTANT If your input module has one or more point or channel-level faults, the input diagnostic subroutines continue to use data from the unfaulted points or channels of that module in comparisons.

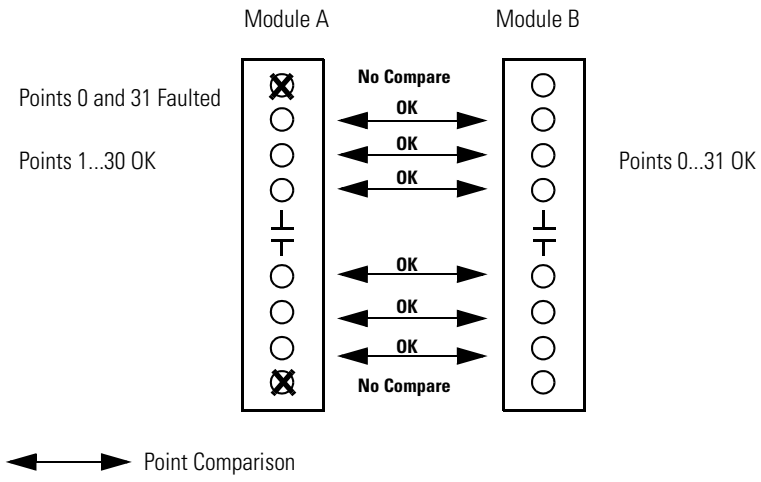
Removing the swing-arm of a 1756-IB32 module results in all points going to zero (low). If you remove a swing-arm, even in a 1oo1 state where a point-level fault exists, all of the unfaulted points go to zero (low).

Then, because the unfaulted points that continue to be compared by the subroutine go to zero (low), a shutdown due to a miscompare occurs.

For more information about repairing or replacing a 1756-IB32 module that has point-level faults, see Replacing a Faulted 1756-IB32 Module on page 121.

- **one module of the pair is faulted** due to a communication fault and the system is operating using only data from the unfaulted module.

1oo1 Due to a Point or Channel Fault

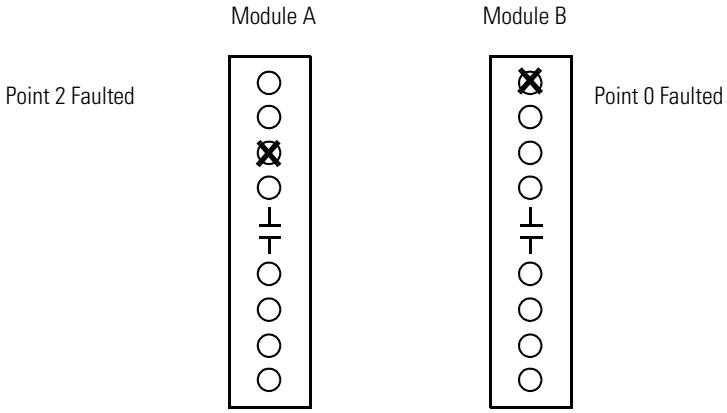


8 / 2011 www.klinkmann.com KLINKMANN

Faulted State

If one or more point or channel-level faults is present on both modules of a pair, a faulted state occurs and the system shutsdown. The faulted state occurs even if the faulted points or channels between module pair are different.

Faulted Due to Faults on Each Module of the Pair



IB32_Diagnostics Subroutine

The 1756-IB32 diagnostic subroutine completes the following tasks when in the states identified.

Normal Operation - 1756-IB32 Module Pair

When in normal operation, the IB32_Diagnostics subroutine carries-out the tasks listed in this table.

System Tasks for 1756-IB32 Normal State

Task	Description
Connection verification	The subroutine verifies that the communication connections are functioning properly. If there is a fault in a module connection, the tags `ConnectionFault_Module_A` and `ConnectionFault_Module_B` indicate the communication fault.
Point-value comparisons	The diagnostic subroutine constantly compares the corresponding point values from the module pair. If a miscompare occurs between the data points, the subroutine initiates the transition test.
Dual-point reconciliation	After the diagnostic subroutine compares the two point values, one from each module of the pair, the two values are reconciled into one bit for use in the main routine.
Initiates transition tests	When a miscompare occurs between points, or when the transition test interval expires, the diagnostic subroutine initiates the transition tests.

Test - 1756-IB32 Module Pair

Transition tests occur at intervals specified by the user or according to the default settings. This table identifies the transition test tags and their default values.

Transition Test Interval Tags

Tag Name	Default Value
`ModulePair_Good_TestInterval`	86400000 (24 hours)
`ModulePair_1oo1_TestInterval`	3600000 (1 hour)

Transition tests are also described in Chapter 2, in the section titled 1756-IB32 DC Input Termination Board and Transition Tests, on page 28.

1oo1 - 1756-IB32 Module Pair

When the module pair is running in a 1oo1 configuration, at least one point of one of the modules in the pair is faulted. The system then runs using data only from the remaining (unfaulted) points of the module and the other unfaulted module.

When the 1756-IB32 module pair is running in a 1oo1 configuration, the diagnostic subroutine carries-out the tasks listed in this table.

System Tasks for 1756-IB32 1oo1 State

Task	Description
Countdown timer starts	When the system begins operating in the 1oo1 state, the diagnostic subroutine starts a timer that when expired, annunciates that the user-defined repair time has elapsed. The repair time is specified in tag `TimeToRun_1oo1`. The system will continue to run in a 1oo1 configuration after the repair time has elapsed. To reset the timer, toggle the `FaultReset` bit.
Transition test frequency increases	When the system is running in a 1oo1 configuration, the diagnostic subroutine carries out transition tests on the remaining module more frequently. The frequency of the transition test is user-defined, however, the default is once per hour. The the transition test frequency is specified in the `ModulePair1oo1_TestInterval` tag.
Module status updated	When the system is operating in a 1oo1 configuration, the IB32_Diagnostics subroutine provides module status information that is useful for troubleshooting the faulted module.

IF16_Diagnostics Subroutine

The 1756-IF16 diagnostic subroutines carry-out these tasks when in the states identified.

Normal Operation - 1756-IF16 Module Pair

When in normal operation, the IF16_Diagnostic subroutine carries-out the tasks listed in this table.

System Tasks for 1756-IF16 Normal State

Task	Description
Connection verification	The subroutine verifies that the communication connections are functioning properly. If there is a fault in the connection to a module, the tags `ConnectionFault_Module_A` and `ConnectionFault_Module_B` indicate the communication faults.
Channel-value comparisons	The diagnostic subroutine constantly compares the corresponding channel values from the module pair. The two channel values, one from each module, must be within the user-defined deadband range of each other. The default deadband range is +/- 5% of the full scaling range.
Dual-channel reconciliation	If the two channels are within the deadband of each other, the system averages the two values and provides a single, reconciled value in a word for use in the main routine. If the two channel values are not within the deadband range, then the diagnostic subroutine initiates a reference test to determine which module of the pair is faulted.
Reference tests initiated	When the two channels of a module pair are not within deadband range of each other, or when the reference test interval expires, the diagnostic subroutine initiates the reference test.

Test - 1756-IF16 Module Pair

Reference tests occur at intervals specified by the user or according to the default settings.

Reference tests are also described in Chapter 2, in the section titled 1756-IF16 Module Pair Reference Tests, on page 34.

1oo1 - 1756-IF16 Module Pair

When the module pair is running in a 1oo1 configuration, at least one channel of one of the modules in the pair is faulted. The system then runs using only data from the remaining (unfaulted) channels of the module and the other unfaulted module.

When the 1756-IF16 module pair is running in a 1oo1 configuration, the diagnostic subroutine carries-out the tasks listed in this table.

System Tasks for 1756-IF16 1oo1 State

Task	Description
Countdown timer starts	When the system begins operating in the 1oo1 state, the diagnostic subroutine starts a timer that when expired, annunciates that the user-defined repair time has elapsed**.** The repair time is specified in tag `TimeToRun_1oo1`. The system will continue to run in a 1oo1 configuration after the repair time has elapsed. The value in the tag `FaultReset` can be toggled to restart the timer.
Reference test frequency increases.	When the system is running in a 1oo1 configuration, the diagnostic subroutine carries out reference tests on the remaining module more frequently. The frequency of the reference test is user-defined, however, the default is once per hour. The the reference test frequency is specified in the `ModulePair_1oo1_TestInterval` tag.
Module status updates.	When the system is operating in a 1oo1 configuration, the IF16_Diagnostics subroutine provides module status information that is useful for troubleshooting the faulted module.

IF16_RefCal Subroutine

In addition to the diagnostic subroutine provided for the 1756-IF16 module pair, another subroutine called IF16_RefCal is also provided.

The IF16_RefCal subroutine carries-out logic that completes these tasks:

- Verifies that all input channels of the 1756-IF16 module pair are reading reference values properly.
- Establishes reference values for each channel that are used by the 1756-IF16 diagnostic subroutine for comparison during the reference test.
- Implements channel scaling values set during the configuration of the 1756-IF16 module pair.

The programming contained in the IF16_RefCal subroutine is carried-out only when initiated in these situations:

- A system start-up, that is, when power is applied or the controller is put into Run mode. At this time, the reference calculations are carried-out on all of the 1756-IF16 module pairs.
- After connections are lost and then re-established on an 1756-IF16 module pair. Only the 1756-IF16 module pair that lost connection will be recalculated.
- When the fault reset button is pressed. The logic provided with the subroutine carries-out a reference calculation on all of the 1756-IF16 module pairs any time fault reset is pressed.

The IF16_RefCal subroutine cannot be edited but it is available for viewing.

8 / 2011 www.klinkmann.com KLINKMANN

OB16D_Diagnostics Subroutine

The 1756-OB16D diagnostic subroutines carry-out the following tasks when in the states identified.

Normal Operation - 1756-OB16D

When in normal operation, the OB16D_Diagnostics subroutine carries-out the tasks listed in this table.

System Tasks for 1756-OB16D Normal State

Task	Description
Connection verification	The subroutine verifies that the communication connections are functioning properly. If a there is a fault in the connection, the tag `ConnectionFault` indicates the communication fault.
Output validation	After the diagnostic condition of the output module pair is determined, the subroutine sends the requested output state to the module pair or an individual module (when in a 1oo1 configuration).
Output data echo and actual output value comparison	The subroutine compares the value returned by the diagnostic output module's data echo to the commanded value of the output bit.
Output module relay control	In the event of a faulted output module, the 1756-OB16D diagnostic subroutine identifies the faulted module and initiates a power disconnect by setting the `Relay_Module` tag to 0. As a result of the Call_Code programming, power is then disconnected from the faulted module using the 1756-OB16D termination board relay.

1oo1 - 1756-OB16D

When the module pair is running in a 1oo1 configuration, one of the modules in the pair has been shut-down and the system is running on information from only the remaining (unfaulted) module. When the 1756-OB16D module pair is running in a 1oo1 configuration, the tasks listed in this table are carried-out.

System Tasks for 1756-OB16D 1oo1 State

Task	Description
Countdown clock	When the system begins operating in the 1oo1 state, the diagnostic subroutine starts a timer that when expired, annunciates that the user-defined repair time has elapsed**.** The repair time is specified in tag `TimeToRun_1oo1`. The system will continue to run in a 1oo1 configuration after the repair time has elapsed. The value in the tag `FaultReset` can be toggled to restart the timer.
Module status	When the system is operating in a 1oo1 configuration, the OB16D_Diagnostics subroutine provides module status information that is useful for troubleshooting the faulted module.

When operating in a 1oo1 state, the pulse test frequency does not increase in the same manner that transition and reference tests do for the input modules. The pulse test continues to be carried-out at the frequency specified in the tag `PulseTest_Interval_PerChnl`.

8 / 2011 www.klinkmann.com KLINKMANN

Data Flow Between Program Elements

It is important for you to understand how data flows in the fault-tolerant program, especially as you complete your system configuration and programming.

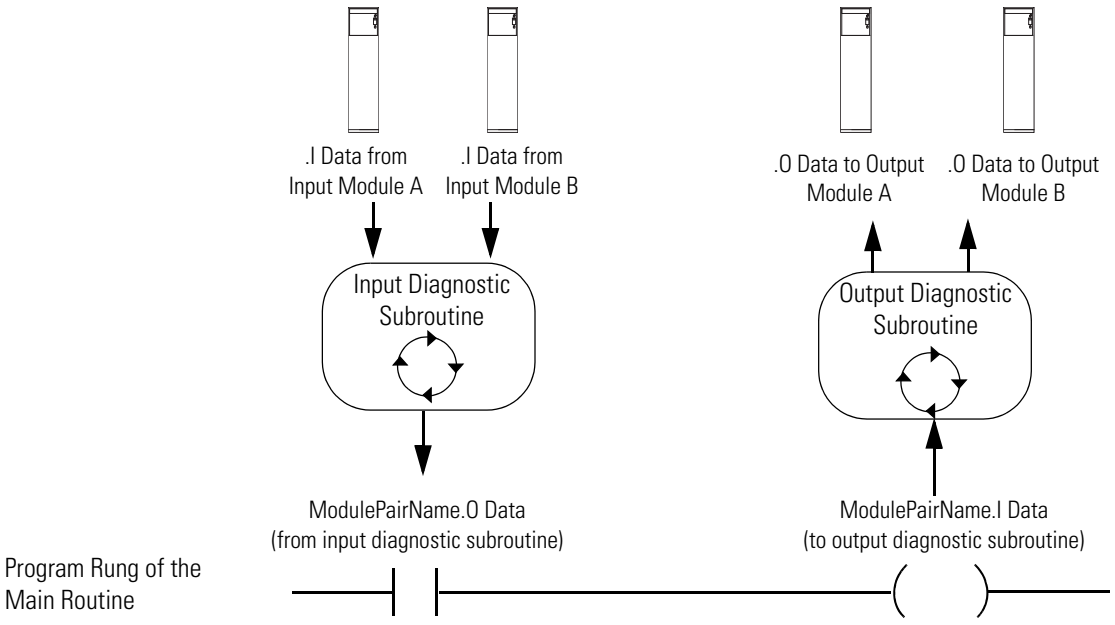
This graphic below provides a view of how data flows and is processed by the fault-tolerant program elements.

Within the fault-tolerant system, data from the both input modules of a pair is processed by the diagnostic subroutines. It is processed and made available in controller tags as one tag that reflects the values provided by both module pairs (called reconciled data).

The data made available by the input diagnostic subroutine is used in programming in the main routine. Based upon the reconciled input value, the system specifies what the value of the outputs are set at.

The output value specified is then processed by the output diagnostic subroutine. The diagnostic subroutine calculates and specifies what the value of each output point should be.

Data and the Typical, Fault-tolerant Input/Output Rung


.I Data from Input Module A
.I Data from Input Module B
.O Data to Output Module A
.O Data to Output Module B
Input Diagnostic Subroutine
Output Diagnostic Subroutine
ModulePairName.O Data (from input diagnostic subroutine)
ModulePairName.I Data (to output diagnostic subroutine)
Program Rung of the Main Routine

The Fault-tolerant Program

Once you understand the elements of the fault-tolerant program and how they function together, you are ready to configure and program your main routine.

Use Chapter 4, Configuring the Fault-tolerant System, and Chapter 5, Programming the Fault-tolerant System, as references when configuring and programming your fault-tolerant system.

Additional Resources

Resource	Description
Logix5000 Common Programming Procedures Programming Manual, publication 1756-PM001	The programming manual describes common techniques and methods for using RSLogix 5000 software to program Logix5000 controllers.
ControlLogix Controllers User Manual, publication 1756-UM001	This manual explains the general use of ControlLogix controllers.
ControlLogix Redundancy System User Manual, publication 1756-UM523	This user manual explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

8 / 2011
www.klinkmann.com KLINKMANN

Configuring the Fault-tolerant System

About This Chapter

This chapter describes procedures for configuring your fault-tolerant system.

Topic	Page
Before You Begin	65
Add the Remote I/O Chassis to the I/O Configuration Tree	67
About System-generated Tags	71
Specifying Diagnostic Subroutine Behavior	72
About ModulePair Tags	72
Create ModulePair Tags	73
Edit ModulePair Tags	76
Editing the 1756-IB32 Call_Code Subroutine	85
Editing the 1756-IF16 Call_Code Subroutine	90
Editing the 1756-OB16D Call_Code Subroutine	95
Next Steps	103
Additional Resources	103

Before You Begin

Before you begin configuring your system using the program supplied by Rockwell Automation, you should prepare your redundant controller chassis and network.

For more information about how to prepare you redundant controller chassis, see the ControlLogix Redundancy System User Manual, publication 1756-UM523.

TIP

We recommend that you configure and program your fault-tolerant system offline.

After you have completed and verified your program, use RSNetWorx for ControlNet software to configure your redundant ControlNet network.

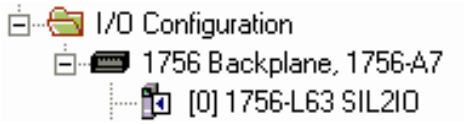
When your ControlNet network is configured, download the program and go online with the controller.

Begin with the Fault-tolerant I/O Program

To begin the configuration of your fault-tolerant system, you must open the fault-tolerant I/O program, titled SIL2_IO_Fault_Tolerant, using RSLogix 5000 software, version 15 or greater.

In this program, a SIL2-certified controller, is present in the configuration tree. Depending on your system, you may need to change the program to specify the controller you are using in your system.

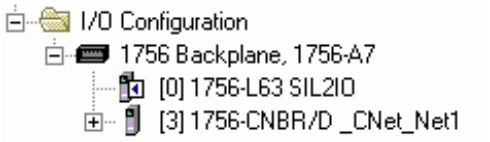
Controller Configuration in Program Supplied by Rockwell Automation



Adding a CNB or CNBR to the Controller Chassis

In order to configure your remote I/O chassis, you must first add a CNB or CNBR module to the chassis configuration provided. Specify the module properties required for your redundant system.

CNBR/D in Controller Chassis



Configuring Remote I/O Chassis

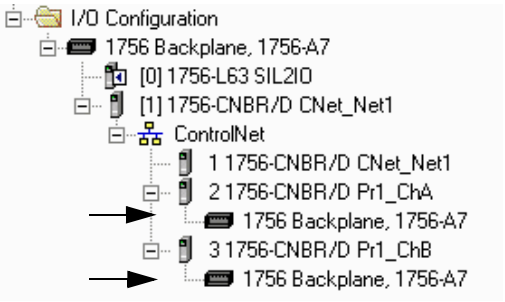
To configure the remote I/O chassis, you must add the remote I/O chassis and their modules to the I/O configuration tree.

Add the Remote I/O Chassis to the I/O Configuration Tree

To add your chassis and remote I/O to the configuration tree, complete these steps.

1. Add two CNB or CNBR modules to the network and specify the Comm Format as None.

   Specify the other module properties according to your system configuration.



2. Add I/O modules to each chassis so the configuration of I/O modules in each chassis is identical.

IMPORTANT The order of the modules in the configuration tree and the module properties of both modules in the pair must be identical.

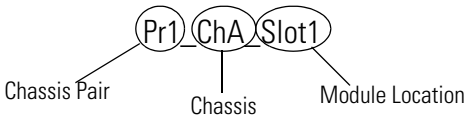
TIP In order to create identical duplicate chassis, you may find it easier to create the first chassis (in this example chassis A) and then copy and paste it into the second chassis (in this example. chassis B).

If you use this method of creating your duplicate chassis, verify that you have edited the parameters of the pasted configuration so that they are specific to that chassis.

TIP

When configuring your I/O modules, use naming conventions that will allow you to easily identify the chassis pair, individual chassis, and module location.

For example, the I/O configuration examples in this manual use the following naming convention.

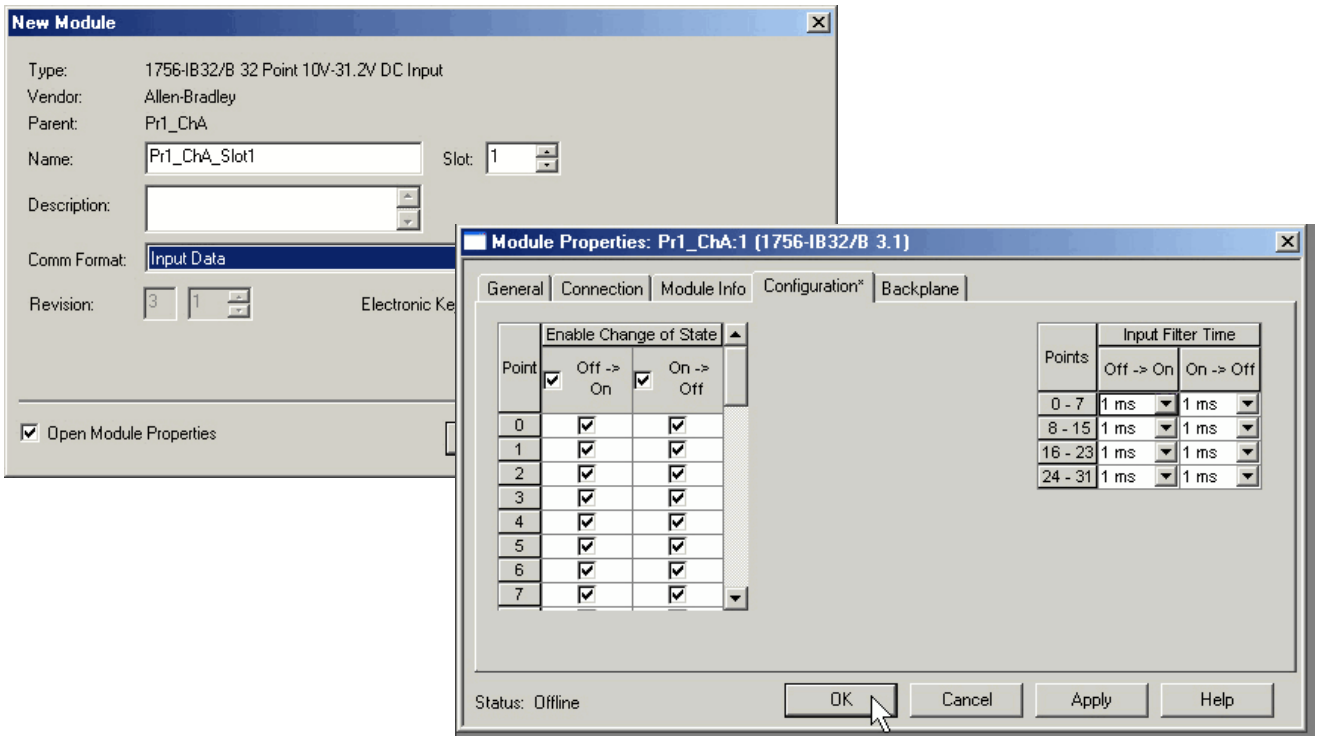


Creating tags with easy-to-understand identifiers helps when programming and troubleshooting the system.

IMPORTANT

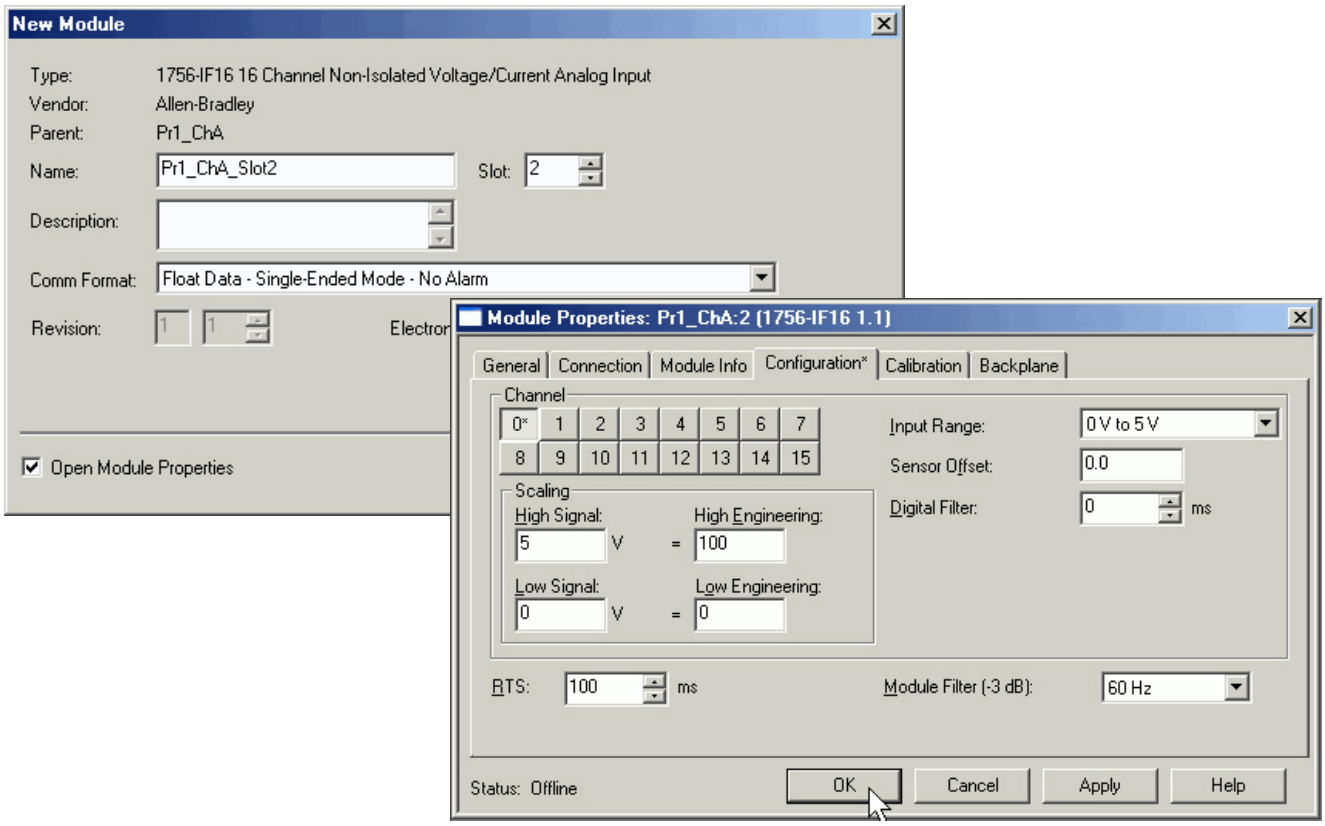
Specify these module properties when adding and configuring I/O modules.

1756-IB32 Module Properties



Property	Value
Comm Format	Input Data
Input Filter Time	Must be identical between the two modules of the pair

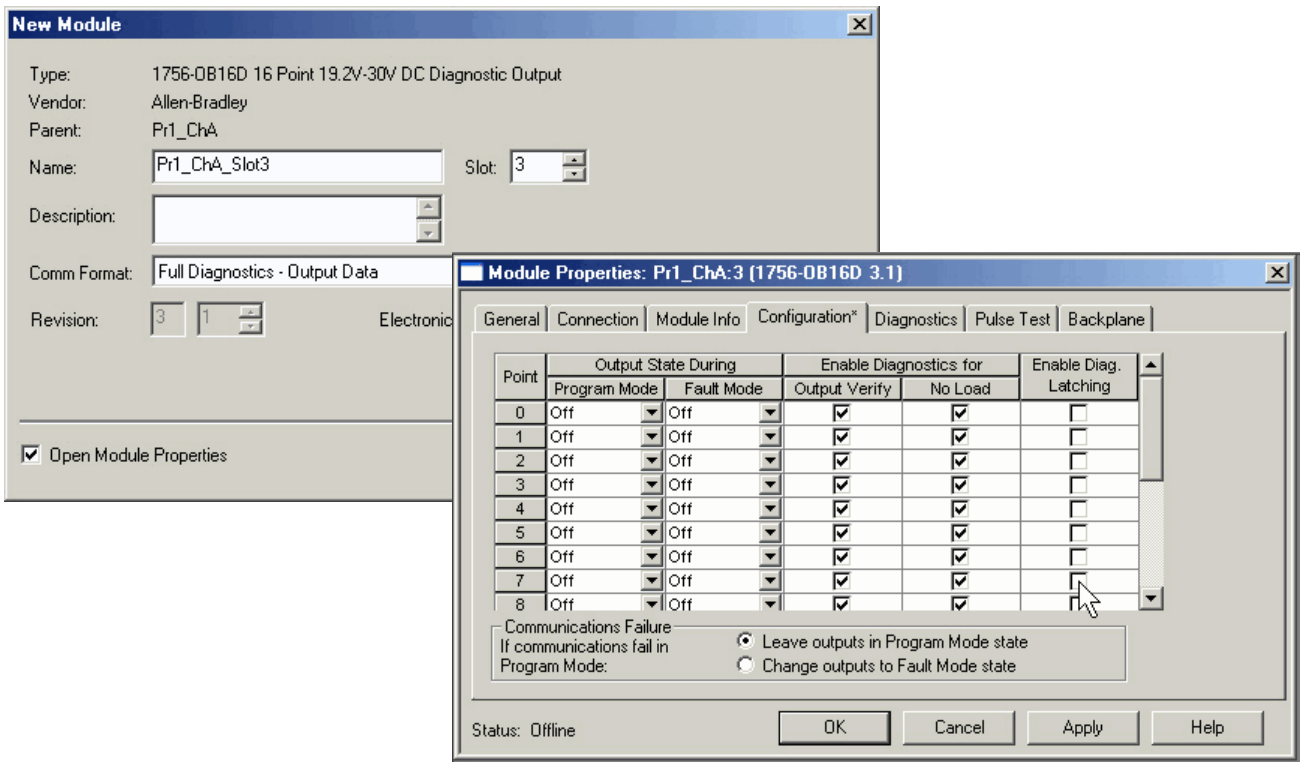
1756-IF16 Module Properties



Property	Value
Comm Format	Float Data -Single-Ended Mode -No Alarm
Input Range	0 V...5 V for each channel (scaling is permitted)

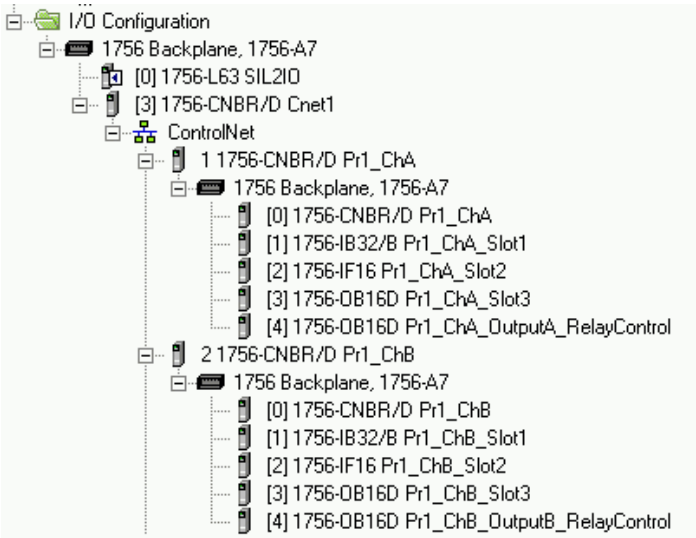
IMPORTANT If you edit the 1756-IF16 module configuration any time after your initial start up, you must press fault reset in order to implement the new configuration parameters.

1756-OB16D Module Properties



Property	Value
Comm Format	Full Diagnostics - Output Data
Enable Diag. Latching	Do not enable (uncheck boxes)

Once your chassis have been configured, your I/O configuration tree should be similar to the one below.



About System-generated Tags

For each module you configure, the system generates tags for the module are created. These tags are also referred to as module-defined tags.

To view these tags, open the Controller Tags folder.

System-generated Tags Resulting From I/O Configuration

Pr1_ChA:1:C	{...}	{...}	AB:1756_DI:C:0
Pr1_ChA:1:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChA:2:C	{...}	{...}	AB:1756_IF8_Flo...
Pr1_ChA:2:I	{...}	{...}	AB:1756_IF8_Flo...
Pr1_ChA:3:C	{...}	{...}	AB:1756_DO:C:0
Pr1_ChA:3:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChA:3:O	{...}	{...}	AB:1756_CNB_S...
Pr1_ChA:4:C	{...}	{...}	AB:1756_DO:C:0
Pr1_ChA:4:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChA:4:O	{...}	{...}	AB:1756_CNB_S...
Pr1_ChA:I	{...}	{...}	AB:1756_CNB_7...
Pr1_ChA:O	{...}	{...}	AB:1756_CNB_7...
Pr1_ChB:1:C	{...}	{...}	AB:1756_DI:C:0
Pr1_ChB:1:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChB:2:C	{...}	{...}	AB:1756_IF8_Flo...
Pr1_ChB:2:I	{...}	{...}	AB:1756_IF8_Flo...
Pr1_ChB:3:C	{...}	{...}	AB:1756_DO:C:0
Pr1_ChB:3:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChB:3:O	{...}	{...}	AB:1756_CNB_S...
Pr1_ChB:4:C	{...}	{...}	AB:1756_DO:C:0
Pr1_ChB:4:I	{...}	{...}	AB:1756_CNB_S...
Pr1_ChB:4:O	{...}	{...}	AB:1756_CNB_S...
Pr1_ChB:I	{...}	{...}	AB:1756_CNB_7...
Pr1_ChB:O	{...}	{...}	AB:1756_CNB_7...

The data in these tags is sensor data from the I/O modules and is used by the diagnostic subroutines (as specified in the JSR instructions of the Call_Codes) to compare point and channel values. The data from the I/O modules is also used when the subroutines complete diagnostic tests and checks.

Specifying Diagnostic Subroutine Behavior

In order to specify the behavior of the diagnostic subroutines, complete these tasks.

Task	Page
Create ModulePair Tags	73
Edit ModulePair Tags	76

About ModulePair Tags

Tags of type ModulePair are user-defined data types created by Rockwell Automation specifically for fault-tolerant SIL2 applications. For each module type (that is 1756-IB32, 1756-IF16, and 1756-OB16D), a ModulePair data type is available.

Once each ModulePair tag is created, a group of tags that are used to specify the behavior in the module pair's diagnostic subroutine are available.

For more information about the tags available for each module pair, see step 2 of the section Create ModulePair Tags.

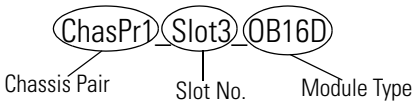
Create ModulePair Tags

1. In the Edit tab of the Controller Tags folder, add a tag for each module pair in the system.

TIP

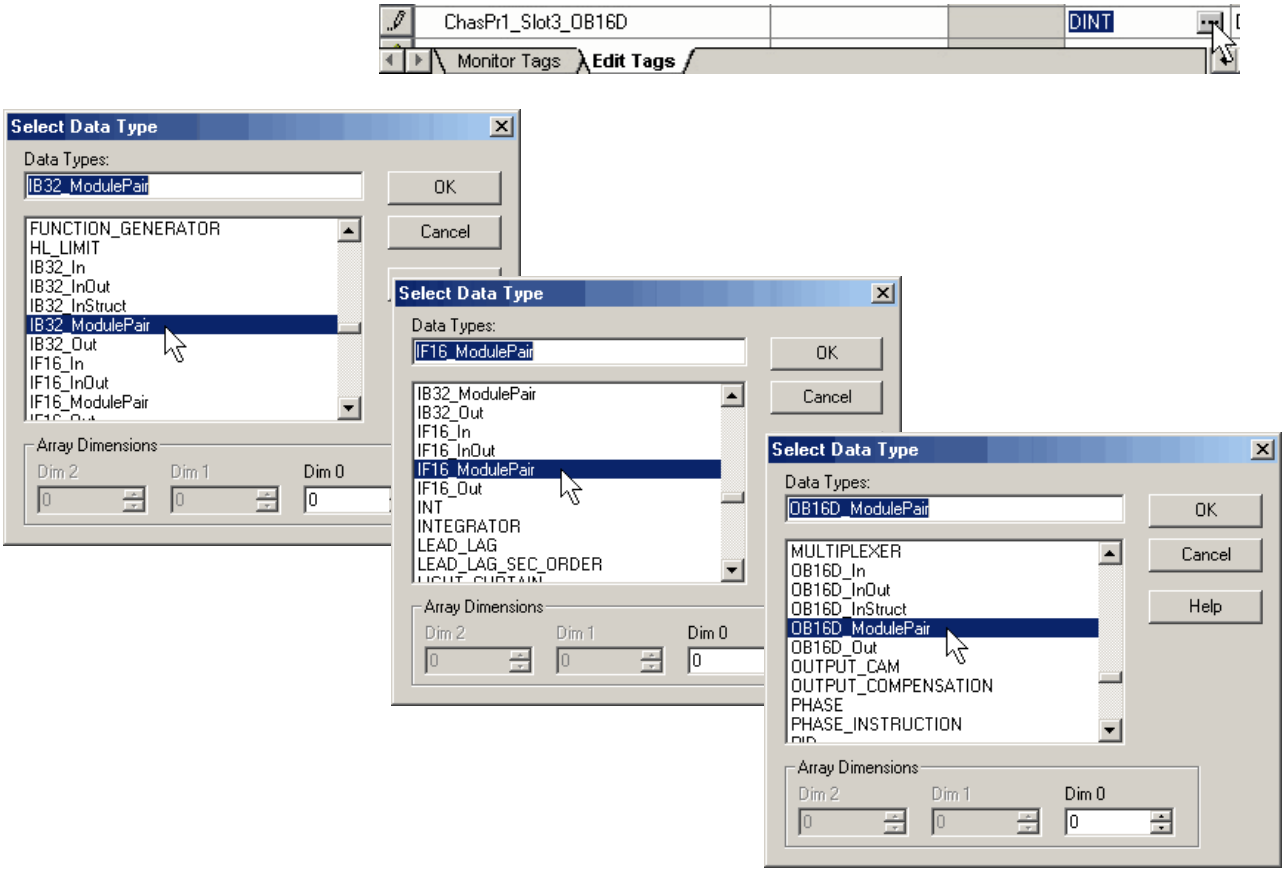
When creating your module pair tags, use naming conventions that will allow you to easily identify the chassis pair, module pair, and module type.

For example, the module pair tag examples in this manual use the following naming convention.



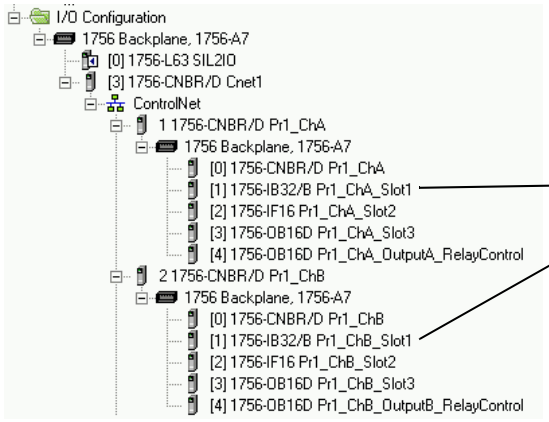
Creating tags with easy-to-understand indentifiers helps when programming and troubleshooting the system.

2. In the Data Type column of each tag, specify the module-specific, ModulePair data type.



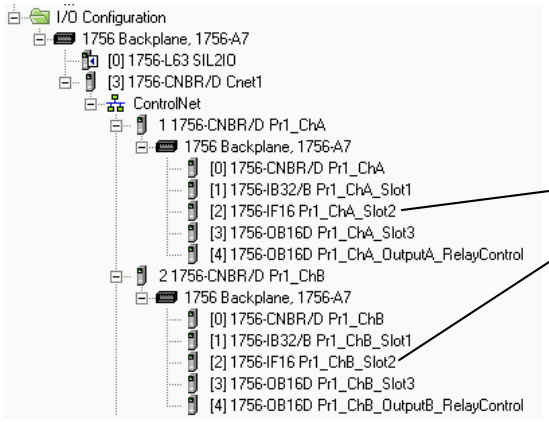
After you have created the tags using the ModulePair data type, these tags and structures result. Each ModulePair tag should correspond to one module pair in your system.

) Configuration Tree

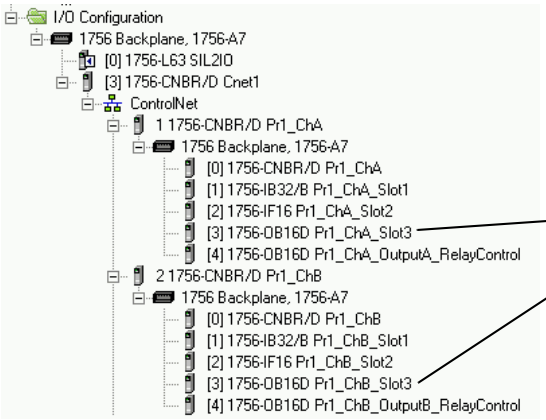


Module Pair Tags

ChasPr1_Slot1_IB32_1			IB32_ModulePair
ChasPr1_Slot1_IB32_1.I			IB32_In
ChasPr1_Slot1_IB32_1.IO			IB32_InOut
ChasPr1_Slot1_IB32_1.O			IB32_Out



ChasPr1_Slot2_IF16			IF16_ModulePair
ChasPr1_Slot2_IF16.I			IF16_In
ChasPr1_Slot2_IF16.IO			IF16_InOut
ChasPr1_Slot2_IF16.O			IF16_Out



ChasPr1_Slot3_OB16D			OB16D_ModulePair
ChasPr1_Slot3_OB16D.I			OB16D_In
ChasPr1_Slot3_OB16D.IO			OB16D_InOut
ChasPr1_Slot3_OB16D.O			OB16D_Out

Some of these tags are used when constructing the main routine, while others are used to specify diagnostic behavior within the subroutines.

Edit ModulePair Tags

After you have created your module pair tags, you must edit the resulting tags in order to specify the behavior of the diagnostic subroutine.

For each type of module pair used, a different group of tag values must be edited. Some of the module pair tags require that values specified in this manual be used. The tags that have specific, required values are described in the sections titled Required 1756-*XXXX* ModulePair Tag Values.

For other module pair tag values, Rockwell Automation recommends values. However, depending on your application, you may choose to use values other than those provided in this manual. These tag values are described in the Recommended 1756-*XXXX* Tag Values sections.

No matter which module pair type you are using, you must enter or edit all of the tag values (required and recommended) described here. Use the section specific to your module pair as a reference when editing the module pair tags.

For section	See page
Editing 1756-IB32 ModulePair Tags	77
Required 1756-IB32 ModulePair Tag Values	78
Recommended 1756-IB32 ModulePair Tag Values	78
Editing 1756-IF16 ModulePair Tags	79
Required 1756-IF16 ModulePair Tag Values	80
Recommended 1756-IF16 ModulePair Tag Values	80
Editing 1756-OB16D ModulePair Tags	82
Required 1756-OB16D ModulePair Tag Values	83
Recommended 1756-OB16D ModulePair Tag Values	83

Editing 1756-IB32 ModulePair Tags

Once the 1756-IB32_ModulePair tags have been generated, these tags specific to the 1756-IB32 module pair result. Located within this group of tags are those you must edit in order to specify system behavior for the 1756-IB32 module pair.

Tag values required. See the Required 1756-IB32 ModulePair Tag Values for values.

Tag values recommended. See the Recommended 1756-IB32 ModulePair Tag Values for recommended values and descriptions.

Do not edit these tags values - they are set by main routine and diagnostic subroutine when the program is running.

⊟-ChasPr1_Slot1_IB32		IB32_ModulePair	
⊟-ChasPr1_Slot1_IB32.I		IB32_In	
⊞-ChasPr1_Slot1_IB32.I.Safety_Inputs_Select		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.I.Miscompare_Test_Limit		DINT	Decimal
⊟-ChasPr1_Slot1_IB32.IO		IB32_InOut	
⊞-ChasPr1_Slot1_IB32.IO.ModulePair_Good_TestInte...		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.ModulePair_1oo1_TestInter...		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.TimeToRun_1oo1		TIMER	Decimal
⊞-ChasPr1_Slot1_IB32.IO.TransitionTest_Low_Delay		TIMER	Decimal
⊞-ChasPr1_Slot1_IB32.IO.TransitionTest_High_Delay		TIMER	Decimal
ChasPr1_Slot1_IB32.IO.CircuitReset		BOOL	Decimal
ChasPr1_Slot1_IB32.IO.FaultReset		BOOL	Decimal
ChasPr1_Slot1_IB32.IO.Run_TransitionTest		BOOL	Decimal
ChasPr1_Slot1_IB32.IO.ConnectionFault_Module_A		BOOL	Decimal
ChasPr1_Slot1_IB32.IO.ConnectionFault_Module_B		BOOL	Decimal
⊞-ChasPr1_Slot1_IB32.IO.Chnl_OK_Module_A		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.Chnl_OK_Module_B		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.ChnlFlt_StuckAtOne_Modul...		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.ChnlFlt_StuckAtOne_Modul...		DINT	Decimal
⊞-ChasPr1_Slot1_IB32.IO.Chnl_Miscompare_Status		DINT	Decimal
⊟-ChasPr1_Slot1_IB32.O		IB32_Out	
⊞-ChasPr1_Slot1_IB32.O.Data		DINT	Decimal
ChasPr1_Slot1_IB32.O.ModulePair_Good		BOOL	Decimal
ChasPr1_Slot1_IB32.O.ModulePair_1oo1		BOOL	Decimal
ChasPr1_Slot1_IB32.O.ModulePair_Faulted		BOOL	Decimal
ChasPr1_Slot1_IB32.O.Module_A_Faulted		BOOL	Decimal
ChasPr1_Slot1_IB32.O.Module_B_Faulted		BOOL	Decimal
⊞-ChasPr1_Slot1_IB32.O.Run_1oo1_Countdown		DINT	Decimal

For more information about the tags generated by the ModulePair data type, see Appendix A on page 105.

You must specify both the required and recommend values for certain tags as described here.

Required 1756-IB32 ModulePair Tag Values

In this tag for the 1756-IB32 module pair, the value listed must be specified for each point.

Tag Name	Description	Value
I.Safety_Inputs_Select	Any 1756-IB32 module pair inputs used in the fault-tolerant system are designated as safety inputs.	1 at each point used 0 at unused points(1)

(1) Points of the 1756-IB32 module pair not used in the fault-tolerant system and not specified as safety inputs cannot be used for any other purpose.

Recommended 1756-IB32 ModulePair Tag Values

In these tags, the values listed are recommended but not required. You may choose to alter these values to suit your application, however, **you must enter a value for each of the tags listed**.

Tag Name	Description	Value
I.Miscompare_Test_Limit	The number of subsequent program scans where a miscompare between points may occur before a fault is registered. The value of four is strongly recommended in order to avoid nuisance trips as well as to provide a timely safety response. If you choose to specify a value lower than four, your system may experience nuisance trips. However, you may choose to lower the value in order to decrease amount of time between a fault and the system response. Setting a value larger then four is not recommended as the response to a fault may be too long for most safety applications.	4
IO.ModulePair_GoodTestInterval	Time, in ms, between transition tests when no module faults are present.	86400000 (24 hours)
IO.ModulePair_1oo1TestInterval	Time, in ms, between transition tests when the system is running in a 1oo1 configuration.	3600000 (1 hour)
IO.TimetoRun_1oo1.PRE	Preset value for 1oo1 countdown timer, in ms.	28800000 (8 hours)
IO.TransitionTest_Low_Delay.PRE(1)	Amount of time, in ms, delayed to allow the inputs to transition from high to low before checking the results of the transition test. The amount of time to delay should be determined by adding your program scan time to the NUT. For example, if your total program scan time is 80 ms and your NUT is 20 ms, you should set your TransitionTest_Low_Delay value to 100 ms.	100
IO.TransitionTest_High_Delay.PRE(1)	Amount of time, in ms, delayed to allow inputs to transition to high before normal operation is resumed after a transition test. The amount of time to delay should be determined by adding your program scan time to the NUT. For example, if your total program scan time is 80 ms and your NUT is 20 ms, you should set your TransitionTest_Low_Delay value to 100 ms.	100

(1) When specifying your TransitionTest_Low_Delay and TransitionTest_High_Delay values, remember that the system is functioning on the last-known verified data during these periods. If an input connected to the module pair changes (for example, if an E-stop is pressed), it will not be processed until the total time of these two values has expired and the system has stopped using the last-known verified data.

Editing 1756-IF16 ModulePair Tags

Once the 1756-IF16_ModulePair tags have been generated, these tags specific to the 1756-IF16 module pair result. Located within this group of tags are those you must edit in order to specify system behavior for the 1756-IF16 module pair.

Tag value required. See the Required 1756-IF16 ModulePair Tag Values for value.

Tag values recommended. See the Recommended 1756-IF16 ModulePair Tag Values for recommended values and descriptions.

Do not edit these tag values - they are set by the main routine and diagnostic subroutine when the program is running.

ChasPr1_Slot2_IF16		IF16_ModulePair	
ChasPr1_Slot2_IF16.I		IF16_In	
ChasPr1_Slot2_IF16.I.Safety_Inputs_Select		DINT	Decimal
ChasPr1_Slot2_IF16.I.ChnlCompare_Deadband		REAL[16]	Float
ChasPr1_Slot2_IF16.I.ReferenceTest_Deadband		REAL[16]	Float
ChasPr1_Slot2_IF16.I.ChnlValues_at_Fault		REAL[16]	Float
ChasPr1_Slot2_IF16.I.Miscompare_Test_Limit		DINT	Decimal
ChasPr1_Slot2_IF16.IO		IF16_InOut	
ChasPr1_Slot2_IF16.IO.ModulePair_Good_TestInterval		DINT	Decimal
ChasPr1_Slot2_IF16.IO.ModulePair_1oo1_TestInterval		DINT	Decimal
ChasPr1_Slot2_IF16.IO.TimeToRun_1oo1		TIMER	Decimal
ChasPr1_Slot2_IF16.IO.SwitchToRefValue_Delay		TIMER	Decimal
ChasPr1_Slot2_IF16.IO.SwitchToSignal_Delay		TIMER	Decimal
ChasPr1_Slot2_IF16.IO.CircuitReset		BOOL	Decimal
ChasPr1_Slot2_IF16.IO.FaultReset		BOOL	Decimal
ChasPr1_Slot2_IF16.IO.Run_ReferenceTest		BOOL	Decimal
ChasPr1_Slot2_IF16.IO.ConnectionFault_Module_A		BOOL	Decimal
ChasPr1_Slot2_IF16.IO.ConnectionFault_Module_B		BOOL	Decimal
ChasPr1_Slot2_IF16.IO.Chnl_OK_Module_A		DINT	Decimal
ChasPr1_Slot2_IF16.IO.Chnl_OK_Module_B		DINT	Decimal
ChasPr1_Slot2_IF16.IO.ChnlFlt_RefTest_Module_A		DINT	Decimal
ChasPr1_Slot2_IF16.IO.ChnlFlt_RefTest_Module_B		DINT	Decimal
ChasPr1_Slot2_IF16.IO.Chnl_Miscompare_Status		DINT	Decimal
ChasPr1_Slot2_IF16.O		IF16_Out	
ChasPr1_Slot2_IF16.O.Data		REAL[16]	Float
ChasPr1_Slot2_IF16.O.ModulePair_Good		BOOL	Decimal
ChasPr1_Slot2_IF16.O.ModulePair_1oo1		BOOL	Decimal
ChasPr1_Slot2_IF16.O.ModulePair_Faulted		BOOL	Decimal
ChasPr1_Slot2_IF16.O.Module_A_Faulted		BOOL	Decimal
ChasPr1_Slot2_IF16.O.Module_B_Faulted		BOOL	Decimal
ChasPr1_Slot2_IF16.O.Run_1oo1_Countdown		DINT	Decimal

For more information about the tags generated by the ModulePair data type, see Appendix A on page 105.

You must specify both the required and recommend values for certain tags as described here.

Required 1756-IF16 ModulePair Tag Values

In this tag for the 1756-IF16 module pair, values must be specified for each channel based upon whether the channel is used or unused.

Tag Name	Description	Value
I.Safety_Inputs_Select	Enter 1 for any analog input channel being used.(1)	1 in each channel used 0 in each unused channel

(1) Unused safety input channels cannot be used for any other purposes (that is, they cannot be used as nonfault-tolerant I/O channels). We recommend that you configure unused channels for voltages of 0…5V and then jumper or ground unused channels to keep channel values within range.

Recommended 1756-IF16 ModulePair Tag Values

In these tags, the values listed are recommended but not required. You may choose to alter these values to suit your application, however, **you must enter a value for each of the tags listed**.

Tag Name	Description	Value
I.ChnlCompare_Deadband[16](1)	Defines the +/- deadband when the same two channels of the pair are compared during normal operation. The value is entered as a percentage of the engineering or scaled units. For example, in an application where: • High Voltage = 5 V • Low Voltage = 0 V • High Engineering = 200 • Low Engineering = 0 Defining a channel comparison deadband of 0.05 results in the channel comparison being considered a match if the values are within 10 units of each other.	0.05 (at each channel), that is 5%
I.ReferenceTest_Deadband[16](1)	Defines the +/- deadband when, during a reference test, the channel value is compared to the reference voltages. The value is entered as a percentage of the engineering or scaled units. For example, in an application where: • High Voltage = 5 V • Low Voltage = 0 V • High Engineering = 200 • Low Engineering = 0 Defining a channel comparison deadband of 0.05 results in a the channel comparison being considered a match if the values are within 10 units of each other.	0.05 (at each channel), that is 5%
I.ChnlValues_at_Fault[16]	Sets the channel values that are used by fault-tolerant system in the event of both modules of the pair faulting. These values should be entered in engineering units.	0

Tag Name	Description	Value
I.Miscompare_Test_Limit	The number of subsequent program scans where a miscompare between points may occur before a fault is registered. The value of four is strongly recommended in order to avoid nuisance trips as well as provide a timely safety response. If you choose to specify a value lower than four, your system may experience nuisance trips. However, you may choose to lower the value in order to decrease amount of time between a fault and the system response. Setting a value larger then four is not recommended as the response to a fault may be too long for most safety applications.	4
IO.ModulePair_GoodTestInterval.PRE	Time, in ms, between transition tests when no module faults are present.	86400000 (24 hours)
IO.ModulePair_1oo1TestInterval.PRE	Time, in ms, between transition tests when the system is running in a 1oo1 configuration.	3600000 (1 hour)
IO.TimetoRun_1oo1.PRE	Preset value for 1oo1 countdown timer, in ms.	28800000 (8 hours)
IO.SwitchToRefValue_Delay.PRE[2]	Amount of time, in ms, delayed to allow the inputs to transition to the reference values before checking the results of the reference test. This value should be equal or greater than your analog module pair's RTS rate.	500
IO.SwitchToSignal_Delay.PRE[1]	Amount of time, in ms, delayed to allow the inputs to transition to the field signal values before normal operation is resumed. This value should be equal or greater than your analog module pair's RTS rate.	500

(1) If changes are made to the ChnlCompare_Deadband or to the ReferenceTest_Deadband tag values after the initial fault-tolerant program is downloaded to and running on the controller, then you must press fault-reset so that the IF16_RefCal subroutine is carried out and the new deadband values are implemented. The changes to these tags are not implemented into the program until the IF16_RefCal subroutine is run.

(2) When specifying your SwitchToRef_Delay and SwitchToSignal_Delay values, remember that the system is functioning on the last-known verified data during these periods. If an input connected to the module pair changes, it will not be processed until the total time of these two values has expired and the system has stopped using the last-known verified data.

8 / 2011 www.klinkmann.com KLINKMANN

Editing 1756-OB16D ModulePair Tags

Once the 1756-OB16D_ModulePair tags have been generated, these tags specific to the 1756-OB16D module pair result. Located within this group of tags are those you must edit in order to specify system behavior for the 1756-OB16D module pair.

Tag values required. See the Required 1756-OB16D ModulePair Tag Values for values.

Tag values recommended. See the Recommended 1756-OB16D ModulePair Tag Values for recommended values and descriptions.

Tag values required. See the Required 1756-OB16D ModulePair Tag Values for these values.

Do not edit these tag values - they are set by the main routine and diagnostic subroutine when the program is running.

ChasPr1_Slot3_OB16D	{...}	{...}	OB16D_ModulePair
ChasPr1_Slot3_OB16D.I	{...}	{...}	OB16D_In
ChasPr1_Slot3_OB16D.I.Safety_Outputs_Select	0		DINT
ChasPr1_Slot3_OB16D.I.Data	0		DINT
ChasPr1_Slot3_OB16D.IO	{...}	{...}	OB16D_InOut
ChasPr1_Slot3_OB16D.IO.PulseTest_Chnl_Select	0		DINT
ChasPr1_Slot3_OB16D.IO.PulseTest_Interval_PerChnl	{...}	{...}	TIMER
ChasPr1_Slot3_OB16D.IO.TimeToRun_1oo1	{...}	{...}	TIMER
ChasPr1_Slot3_OB16D.IO.OneShot_Bits	0		DINT
ChasPr1_Slot3_OB16D.IO.PulseTest_Settings	{...}	{...}	SINT[10]
ChasPr1_Slot3_OB16D.IO.PulseTestResult_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.PulseTestResult_Module_B	0		DINT
ChasPr1_Slot3_OB16D.IO.CircuitReset	0		BOOL
ChasPr1_Slot3_OB16D.IO.FaultReset	0		BOOL
ChasPr1_Slot3_OB16D.IO.Run_PulseTest	0		BOOL
ChasPr1_Slot3_OB16D.IO.ConnectionFault_Module_A	0		BOOL
ChasPr1_Slot3_OB16D.IO.ConnectionFault_Module_B	0		BOOL
ChasPr1_Slot3_OB16D.IO.Chnl_OK_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_OK_Module_B	0		DINT
ChasPr1_Slot3_OB16D.IO.ChnlFlt_PulseTest_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.ChnlFlt_PulseTest_Module_B	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_Grounded_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_Grounded_Module_B	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_HWFail_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_HWFail_Module_B	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_NoLoadOrDCV_Module_A	0		DINT
ChasPr1_Slot3_OB16D.IO.Chnl_NoLoadOrDCV_Module_B	0		DINT
ChasPr1_Slot3_OB16D.O	{...}	{...}	OB16D_Out
ChasPr1_Slot3_OB16D.O.ModulePair_Good	0		BOOL
ChasPr1_Slot3_OB16D.O.ModulePair_1oo1	0		BOOL
ChasPr1_Slot3_OB16D.O.ModulePair_Faulted	0		BOOL
ChasPr1_Slot3_OB16D.O.Module_A_Faulted	0		BOOL
ChasPr1_Slot3_OB16D.O.Module_B_Faulted	0		BOOL
ChasPr1_Slot3_OB16D.O.Run_1oo1_Countdown	0		DINT
ChasPr1_Slot3_OB16D.O.Relay_Module_A	0		BOOL
ChasPr1_Slot3_OB16D.O.Relay_Module_B	0		BOOL

For more information about the tags generated by the ModulePair data type, see Appendix A on page 105.

You must specify both the required and recommend values for certain tags as described here.

Required 1756-OB16D ModulePair Tag Values

These values are required for 1756-OB16D module pair tags.

Tag Name	Description	Value
I.Safety_Outputs_Select	For fault-tolerant I/O, all 1756-OB16D module pair outputs are designated as safety outputs.	1 for all points, used or unused
IO.PulseTest_Settings[4]	Sets the maximum pulse test width and is specified in 100 µs increments.	20 (2 ms)
IO.PulseTest_Settings[8]	Sets the amount of time, in 100 µs increments, for the delay between the end of the pulse test and the declaration of a fault.	20 (2 ms)

Recommended 1756-OB16D ModulePair Tag Values

In these tags, the values listed are recommended but not required. You may choose to alter these values to suit your application, however, **you must enter a value for each of the tags listed**.

Tag Name	Description	Value
IO.PulseTest_Chnl_Select	Use to enable or disable the execution of pulse tests on points of the output module pair.(1)	1 = Pulse test enabled 0 = Pulse test disabled
IO.PulseTest_Interval_PerChnl.PRE	Time, in ms, between pulse tests on individual output points. The total time it takes for pulse tests to be carried-out on all points of the module pair is this value multiplied the number of outputs. This is true even when pulse tests are disabled for any of the points. For example, when the 5 s is the PulseTest_Interval_PerChnl value, the total time required for all of the outputs to be pulse tested is 80 seconds (that is, 16 points x 5 s = 80 s).	5000 (5 s)
IO.TimeToRun_1oo1.PRE	Preset value for the 1oo1 countdown timer, in ms.	28800000 (8 hour)

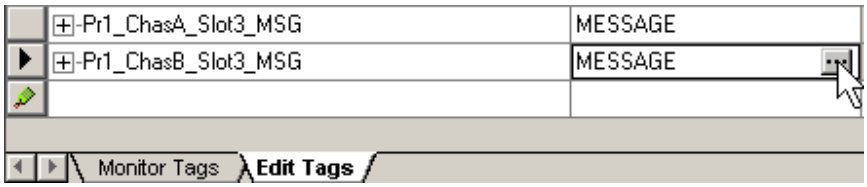
(1) Pulse tests must be disabled for outputs used to trigger diagnostic tests (that is, transition or reference tests) on input module pairs and outputs used to control relays on output termination boards.

Adding MESSAGE Tags

The OB16D_Call_Code subroutine uses MSG instructions to initiate the pulse tests for the module pair. The MSG instructions require the use of MESSAGE tags. Later in the configuration, you will edit the MSG instructions to use the tags you create here.

You must add a MESSAGE tag for each 1756-OB16D module of each module pair in your system. For example, if you have three 1756-OB16D module pairs in your system, you need six tags of the MESSAGE type.

To add a MESSAGE tag, create the tag in the Controller Tags list and specify the MESSAGE data type.



Editing the Call_Code Subroutines

You must edit the Call_Code subroutines to call the diagnostic subroutines for each module pair in your system. This section describes the steps required to edit the Call_Code subroutines for each type of module pair (that is, the 1756-IB32, 1756-IF16, and 1756-OB16D module pairs).

To edit the Call_Code subroutines, simply copy and paste the sample rungs provided and specify the ModulePair tags that correspond to the module pairs in your system.

See the section specific to your module pair type for information about editing the Call_Code Subroutines.

For ModulePair type	See
1756-IB32	page 85
1756-IF16	page 90
1756-OB16D	page 95

Editing the 1756-IB32 Call_Code Subroutine

This section describes how to edit the 1756-IB32 Call_Code subroutine for fault-tolerant applications

To edit the 1756-IB32 Call_Code subroutine, complete these tasks.

Task	Page
Copy and Paste a JSR Rung for Each 1756-IB32 Module Pair	85
Edit JSR Parameters for the 1756-IB32 Module Pair	87
Edit Other Rung Elements for the 1756-IB32 Module Pair	88

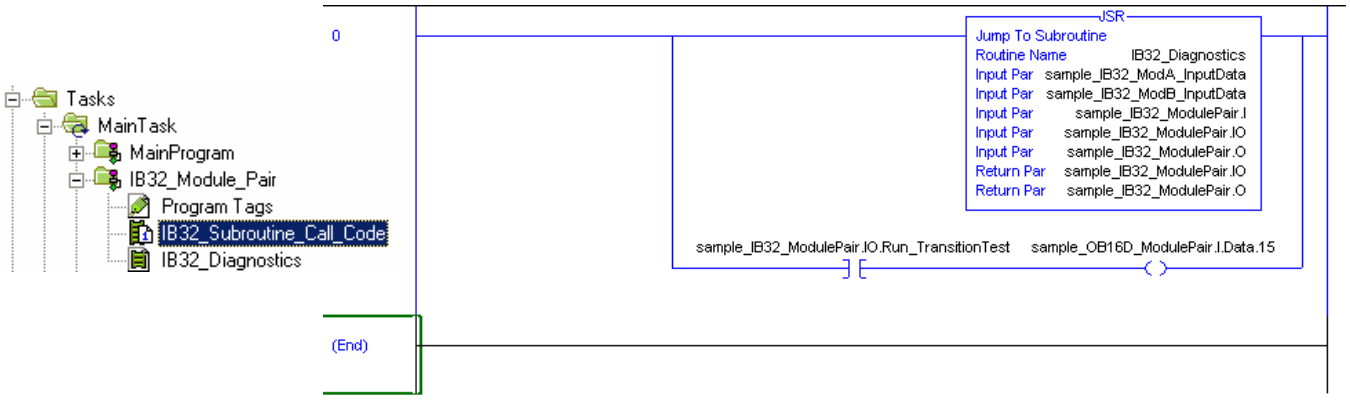
Copy and Paste a JSR Rung for Each 1756-IB32 Module Pair

To add a JSR instruction run for 1756-IB32 module pair, complete the following steps.

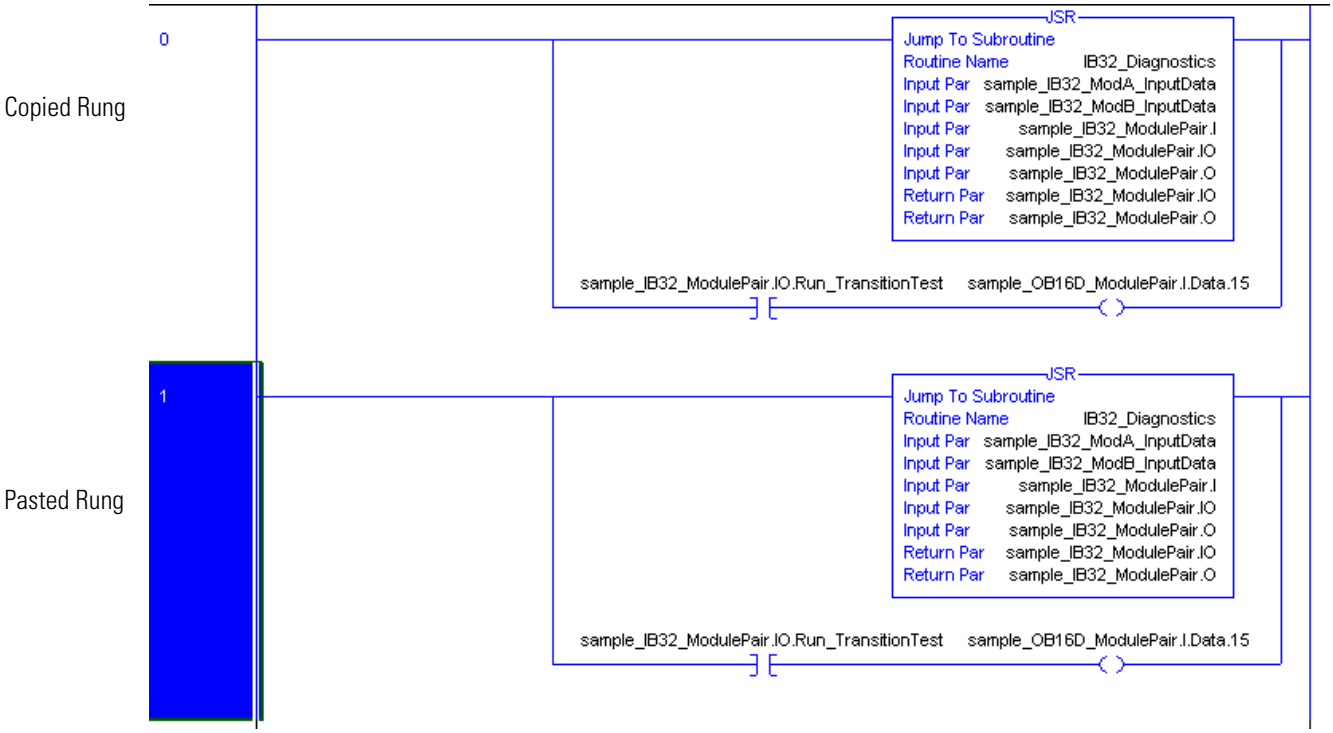
1. Open the IB32_Call_Code routine.

   The example program ladder logic displays.

1756-IB32 Call_Code



2. Copy the rung provided and paste it.



3. Repeat steps 1…2 until there is a JSR instruction rung for every 1756-IB32 input module pair in the system.

After you have created a JSR instruction rung for each input module pair, you must edit the JSR parameters and other elements of the rungs.

Edit JSR Parameters for the 1756-IB32 Module Pair

The JSR instruction for the 1756-IB32 diagnostic routine uses four input parameters and two return parameters. You must edit these parameters so that the tags specific to your 1756-IB32 module pair are used.

Also, remember to edit a JSR instruction for each 1756-IB32 module pair in your system. For example, if your system has four 1756-IB32 module pairs, you must edit each of the four JSR instructions to use parameters specific to one 1756-IB32 module pair.

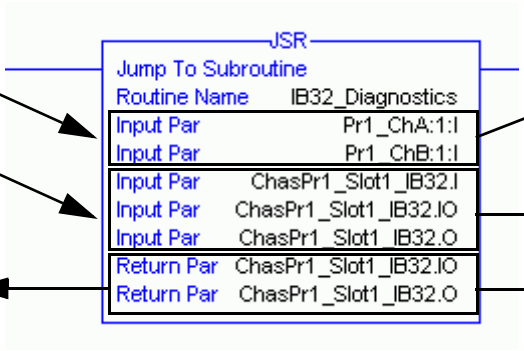
1756-IB32 Module Pair JSR Parameters

About the Data Used

Data from module inputs.

Data specified for system behavior.

Data from diagnostic subroutine.



About the Tags Used

The tags used for these input parameters are system-generated input (.I) tags that were created when you configured your 1756-IB32 modules.

The tags used for these input parameters are the tags that were generated when you created the ModulePair type tags for your 1756-IB32 modules.

The diagnostic subroutine returns data to these tags that were generated when you created the ModulePair type tags.

Use the following table as a reference when editing your 1756-IB32 JSR parameters.

1756-IB32 Module Pair Tags for Use as JSR Parameters

Parameter	Use Tag	Description
Input Par	ModuleAName:*X*:I	System-generated input (.I) tags for module A of the pair.
Input Par	ModuleBName:*X*:I	System-generated input (.I) tags for module B of the pair.
Input Par	ModulePairName.I	ModulePair input (.I) tags that contain module pair behavior data for both modules of the pair.
Input Par	ModulePairName.IO	Tags that contain module pair diagnostic status data for the module pair.
Input Par	ModulePairName.O	Tags containing the reconciled data (that is, resulting data that has been processed by the diagnostic subroutine) for the module pair.

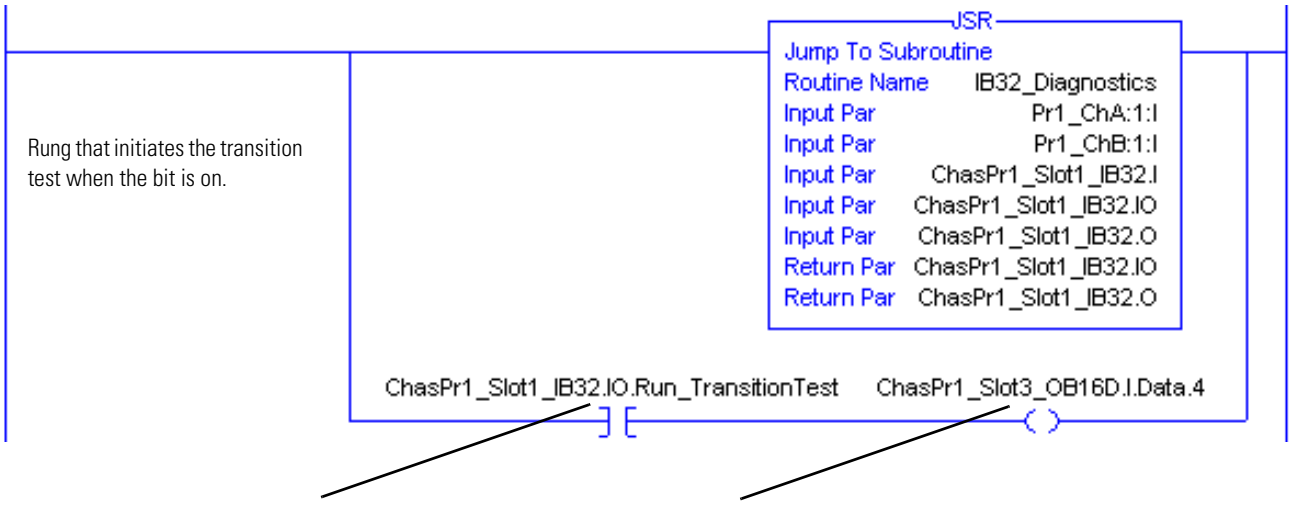
1756-IB32 Module Pair Tags for Use as JSR Parameters

Parameter	Use Tag	Description
Return Par	ModulePairName.IO	Tags that contain module pair diagnostic status data for the module pair.
Return Par	ModulePairName.O	Tags containing the reconciled data (that is, resulting data that has been processed by the diagnostic subroutine) for the module pair.

Edit Other Rung Elements for the 1756-IB32 Module Pair

For each 1756-IB32 module pair, you must also edit the branch associated with the JSR instruction. This branch simply initiates the module pair's transition test when the transition test bit is on.

Other IB32 Subroutine Elements to Edit

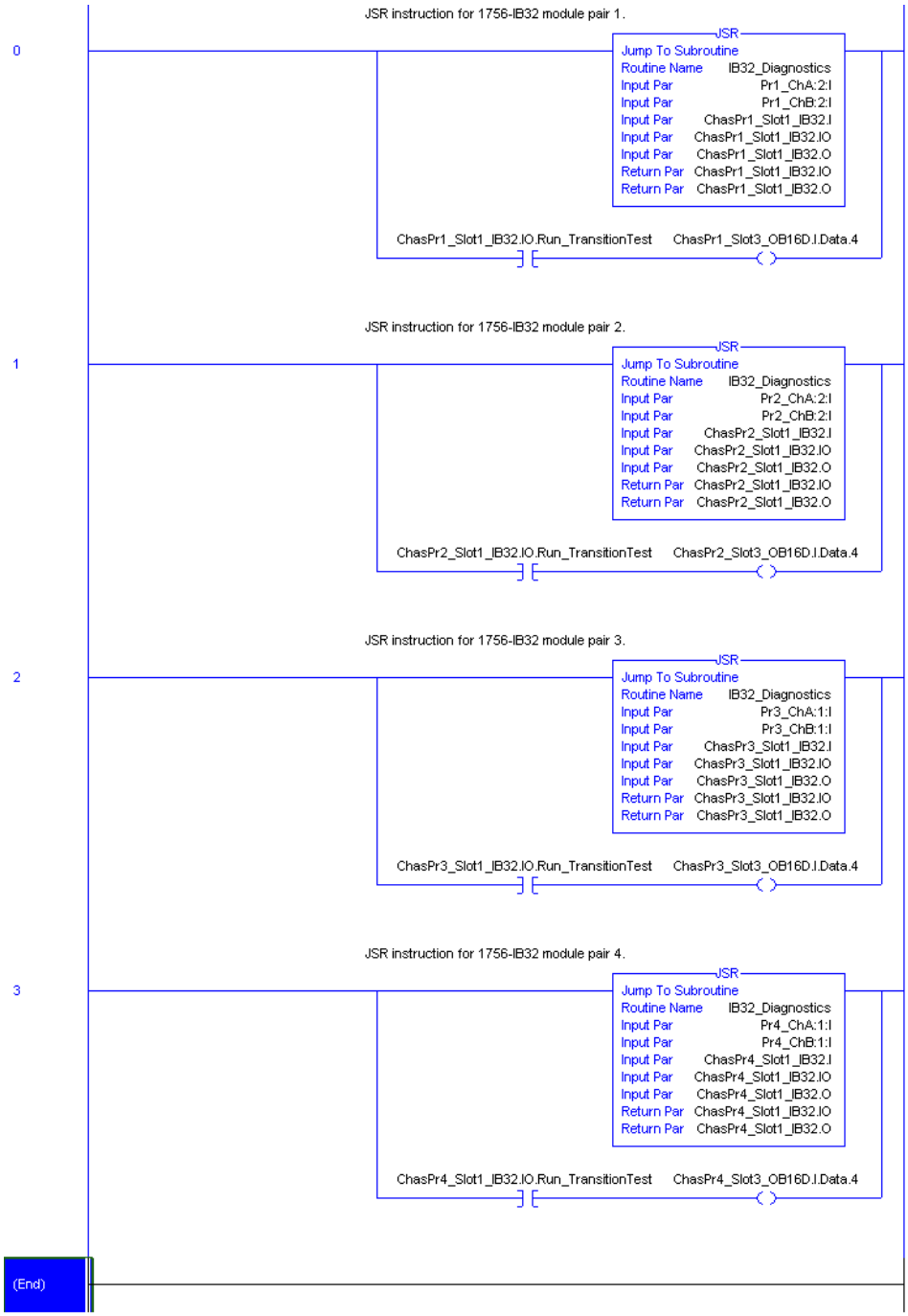

JSR
Jump To Subroutine
Routine Name IB32_Diagnostics
Input Par Pr1_ChA:1:I
Input Par Pr1_ChB:1:I
Input Par ChasPr1_Slot1_IB32.I
Input Par ChasPr1_Slot1_IB32.IO
Input Par ChasPr1_Slot1_IB32.O
Return Par ChasPr1_Slot1_IB32.IO
Return Par ChasPr1_Slot1_IB32.O
Rung that initiates the transition test when the bit is on.
ChasPr1_Slot1_IB32.IO.Run_TransitionTest
ChasPr1_Slot3_OB16D.I.Data.4
If the Run_TransitionTest bit for the module pair is on,an output of the 1756-OB16D module pair that triggers the transition test is turned on.

You must edit the Examine On instruction so that it references the `Run_TransitionTest` tag for the module pair. You must also specify which point of the 1756-OB16D module pair opens the normally-closed relay on the 1756-IB32 termination board. This is how the transition test of the module pair is initiated.

Example of IB32_Call_Code with Completed Edits

This example depicts how the completed IB32_Call_Code subroutine would appear if four 1756-IB32 module pairs were used in the fault-tolerant system.

Example IB32_Call_Code Subroutine with Four Module Pairs



Editing the 1756-IF16 Call_Code Subroutine

This section describes how to edit the 1756-IF16 Call_Code subroutine for fault-tolerant applications.

To edit the 1756-IF16 Call_Code subroutine, complete these tasks:

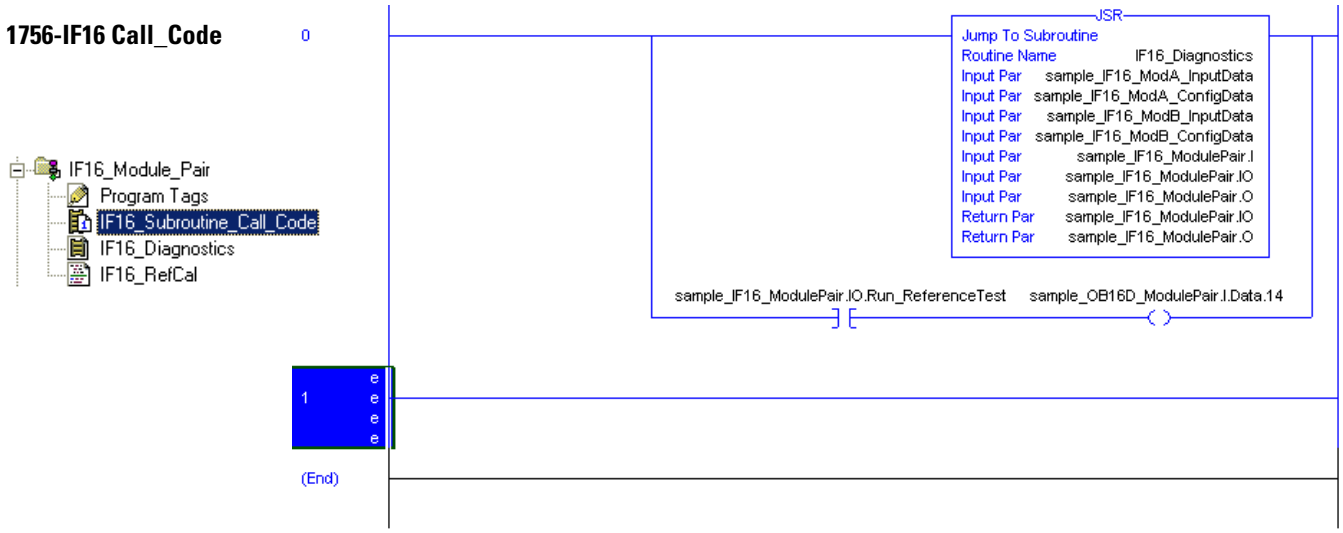
Task	Page
Copy and Paste a JSR Rung for Each 1756-IF16 Module Pair	90
Edit JSR Parameters for the 1756-IF16 Module Pair	92
Edit Other Rung Elements for the 1756-IF16 Module Pair	93

Copy and Paste a JSR Rung for Each 1756-IF16 Module Pair

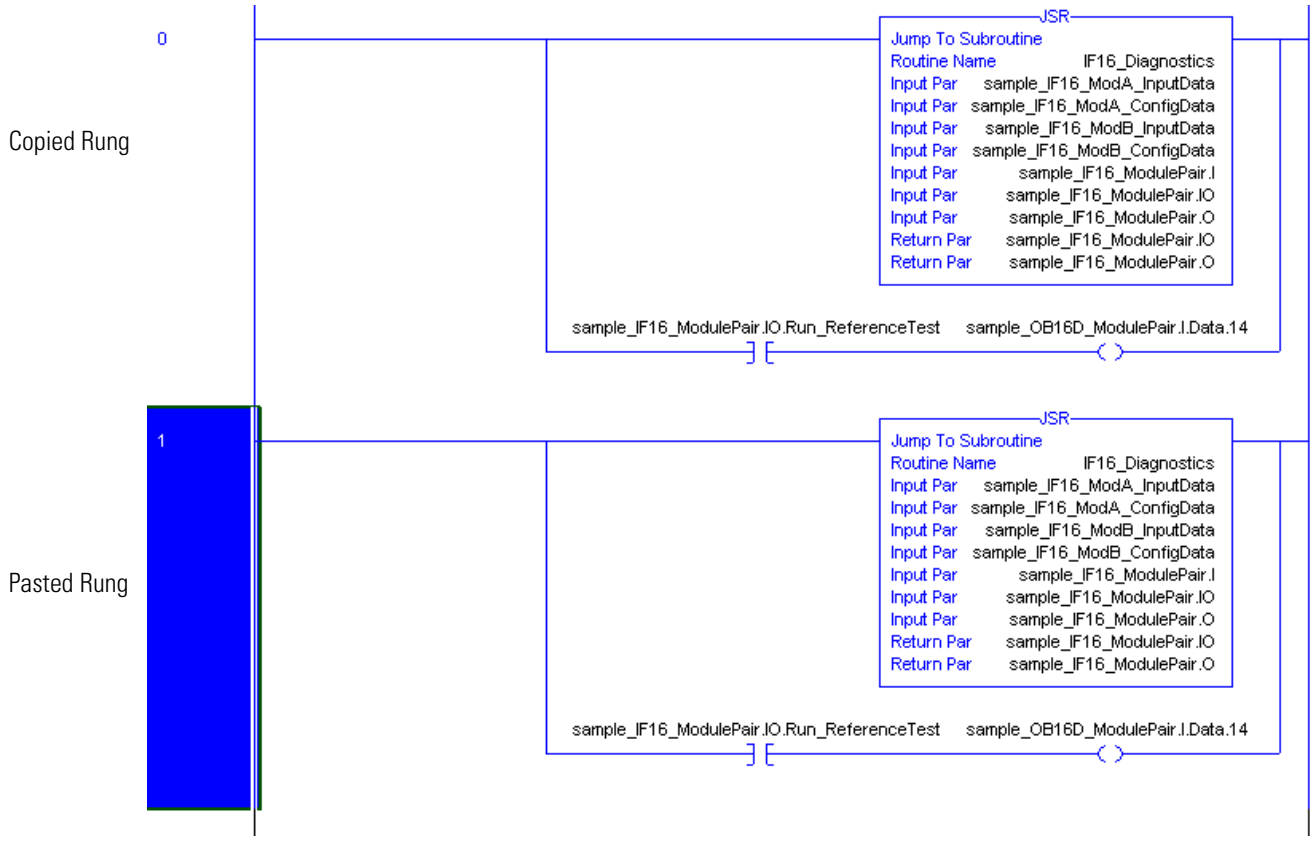
To add a JSR instruction rung for a module pair, complete the following steps.

1. Open the IF16_Call_Code routine.

   The example program ladder logic displays.



2. Copy the rung provided and paste it.



3. Repeat steps 1…2 until there is a JSR instruction rung for every 1756-IF16 input module pair in the system.

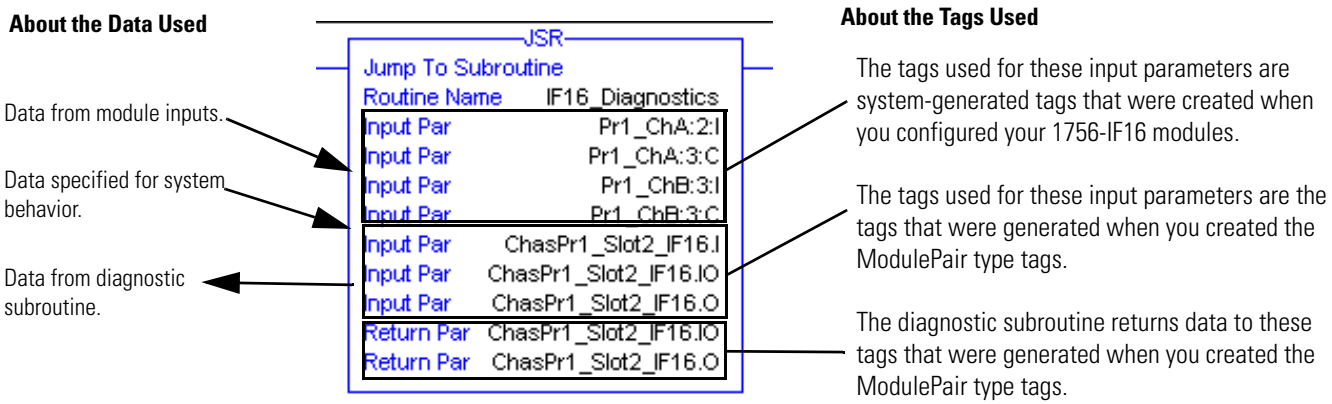
After you have created a JSR instruction rung for each input module pair, you must edit the JSR parameters and other elements of the rungs.

Edit JSR Parameters for the 1756-IF16 Module Pair

The JSR instruction for the 1756-IF16 diagnostic routine uses six input parameters and two return parameters. You must edit these parameters so that the tags specific to your 1756-IF16 module pairs are used.

Also, remember to edit a JSR instruction for each 1756-IF16 module pair in your system. For example, if your system has two 1756-IF16 module pairs, you must edit each of the two JSR instructions to use parameters specific to one 1756-IF16 module pair.

1756-IF16 Module Pair JSR Parameters


About the Data Used
Data from module inputs.
Data specified for system behavior.
Data from diagnostic subroutine.
JSR
Jump To Subroutine
Routine Name IF16_Diagnostics
Input Par Pr1_ChA:2:I
Input Par Pr1_ChA:3:C
Input Par Pr1_ChB:3:I
Input Par Pr1_ChB:3:C
Input Par ChasPr1_Slot2_IF16.I
Input Par ChasPr1_Slot2_IF16.IO
Input Par ChasPr1_Slot2_IF16.O
Return Par ChasPr1_Slot2_IF16.IO
Return Par ChasPr1_Slot2_IF16.O
About the Tags Used
The tags used for these input parameters are system-generated tags that were created when you configured your 1756-IF16 modules.
The tags used for these input parameters are the tags that were generated when you created the ModulePair type tags.
The diagnostic subroutine returns data to these tags that were generated when you created the ModulePair type tags.

Use the following table as a reference when editing your 1756-IF16 JSR parameters.

Tags for Use as 1756-IF16 JSR Parameters

Parameter	Use Tag	Description
Input Par	ModuleAName:*X*:I	System-generated input (.I) tags for module A of the pair.
Input Par	ModuleAName:*X*:C	System-generated configuration (.C) tags for module A of the pair.
Input Par	ModuleBName:*X*:I	System-generated input (.I) tags for module B of the pair.
Input Par	ModuleBName:*X:C*	System-generated configuration (.C) tags for module B of the pair.
Input Par	ModulePairName.I	ModulePair input (I.) tags that contain module pair behavior specification data for both modules of the pair.
Input Par	ModulePairName.IO	Tags that contain module pair diagnostic status data for the module pair.
Input Par	ModulePairName.O	Tags containing the reconciled data (that is, resulting data that has been processed by the diagnostic subroutine) for the module pair.

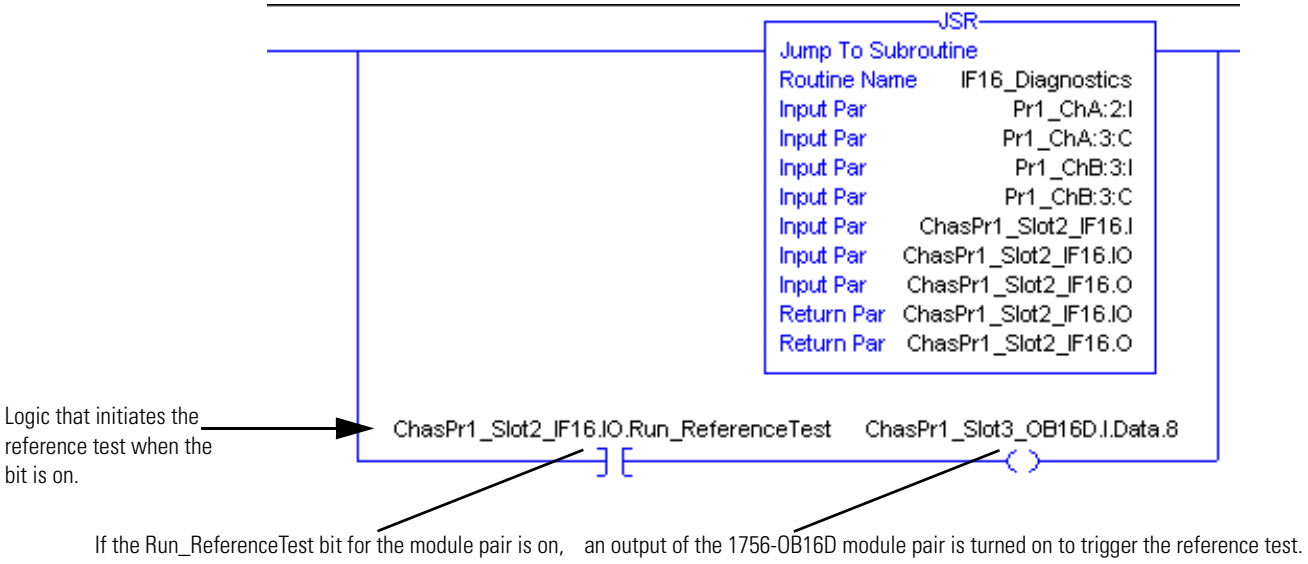
Tags for Use as 1756-IF16 JSR Parameters

Parameter	Use Tag	Description
Return Par	ModulePairName.IO	Tags that contain module pair diagnostic status data for the module pair.
Return Par	ModulePairName.O	Tags containing the averaged input data (that is, resulting data that has been processed by the diagnostic subroutine) for the module pair.

Edit Other Rung Elements for the 1756-IF16 Module Pair

For the 1756-IF16 module pair, you must also edit the corresponding branch. This branch simply initiates the module pair's reference test when the Run_ReferenceTest bit is on.

Other IF16 Subroutine Elements to Edit

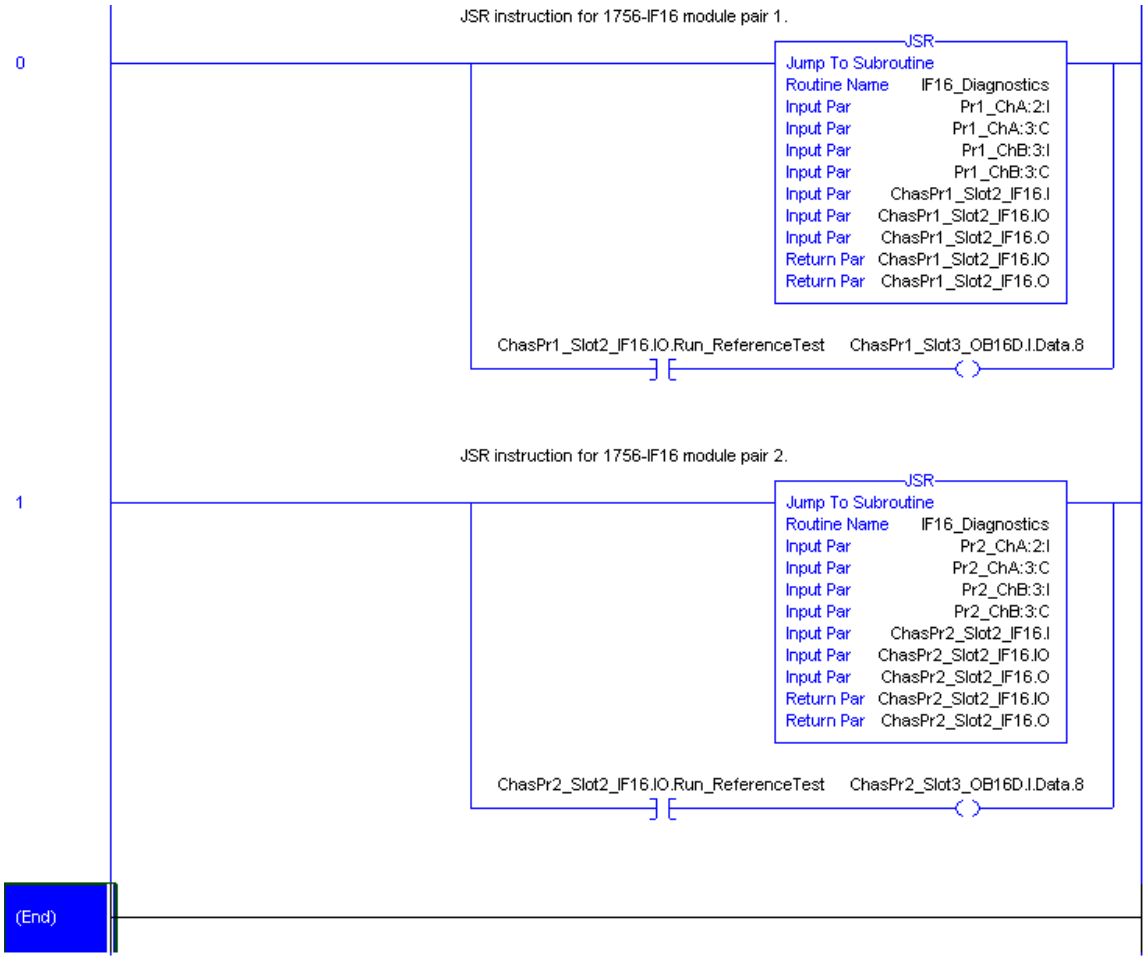

JSR
Jump To Subroutine
Routine Name IF16_Diagnostics
Input Par Pr1_ChA:2:I
Input Par Pr1_ChA:3:C
Input Par Pr1_ChB:3:I
Input Par Pr1_ChB:3:C
Input Par ChasPr1_Slot2_IF16.I
Input Par ChasPr1_Slot2_IF16.IO
Input Par ChasPr1_Slot2_IF16.O
Return Par ChasPr1_Slot2_IF16.IO
Return Par ChasPr1_Slot2_IF16.O
Logic that initiates the reference test when the bit is on.
ChasPr1_Slot2_IF16.IO.Run_ReferenceTest
ChasPr1_Slot3_OB16D.I.Data.8
If the Run_ReferenceTest bit for the module pair is on,
an output of the 1756-OB16D module pair is turned on to trigger the reference test.

Edit the Examine On instruction so that it references the `Run_ReferenceTest` tag for the module pair. You must also specify which point of the 1756-OB16D module pair activates the reference voltages on the analog input termination board.

Example of IF16_Call_Code with Completed Edits

This example depicts how the completed IF16_Call_Code subroutine would appear if two 1756-IF16 module pairs were used in the fault-tolerant system.

Example IF16_Call_Code Subroutine with Two Module Pairs



Editing the 1756-OB16D Call_Code Subroutine

This section describes how to edit the 1756-OB16D Call_Code subroutine for fault-tolerant applications.

To edit the 1756-OB16D Call_Code subroutine, complete these tasks:

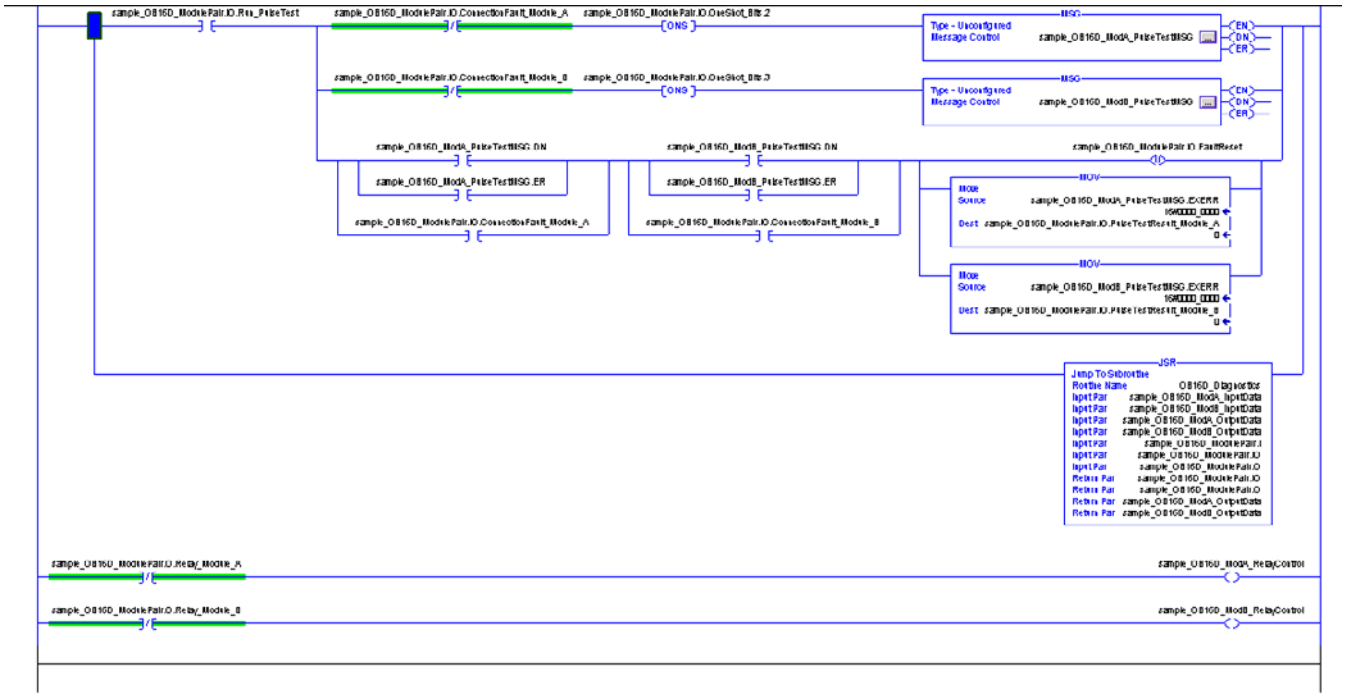
Task	Page
Copy and Paste Rungs for Each 1756-OB16D Module Pair	95
Edit JSR Parameters for the 1756-OB16D Module Pair	102
Edit Elements of the 1756-OB16D Call_Code Routine	97

Copy and Paste Rungs for Each 1756-OB16D Module Pair

To add a JSR instruction for a module pair, complete the following steps.

1. Open the Subroutine_Call_Code routine specific to the module pair type.

   The example program ladder logic displays.



2. Copy rungs 0…2 and paste them below rung 2.

3. Repeat step 2 until each 1756-OB16D module pair has a set of the three rungs in the Call_Code subroutine.

After you have completed creating a set of rungs for each 1756-OB16D module pair, you must then edit each module pairs' set of rungs.

Edit Elements of the 1756-OB16D Call_Code Routine

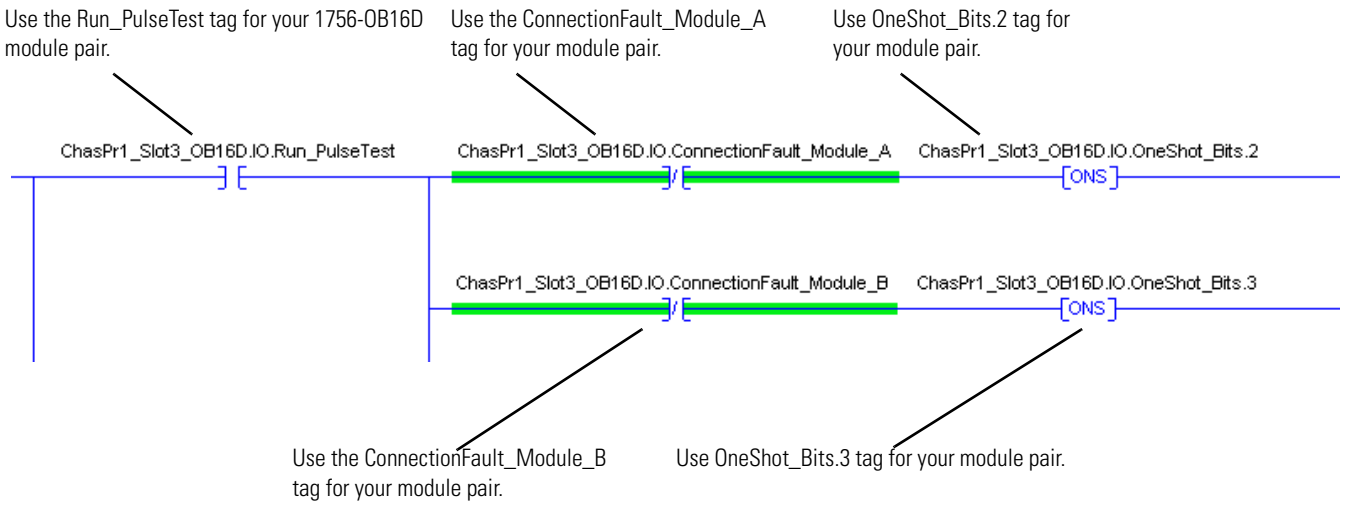
After you have added rung sets for each module pair and entered parameters in each module pair's JSR instruction, you must edit other elements of call_code subroutine program.

Complete these steps to edit the other elements of the call_code subroutine for each 1756-OB16D output module pair.

1. In the first rung, edit the instruction tags as described in the graphics that follow.

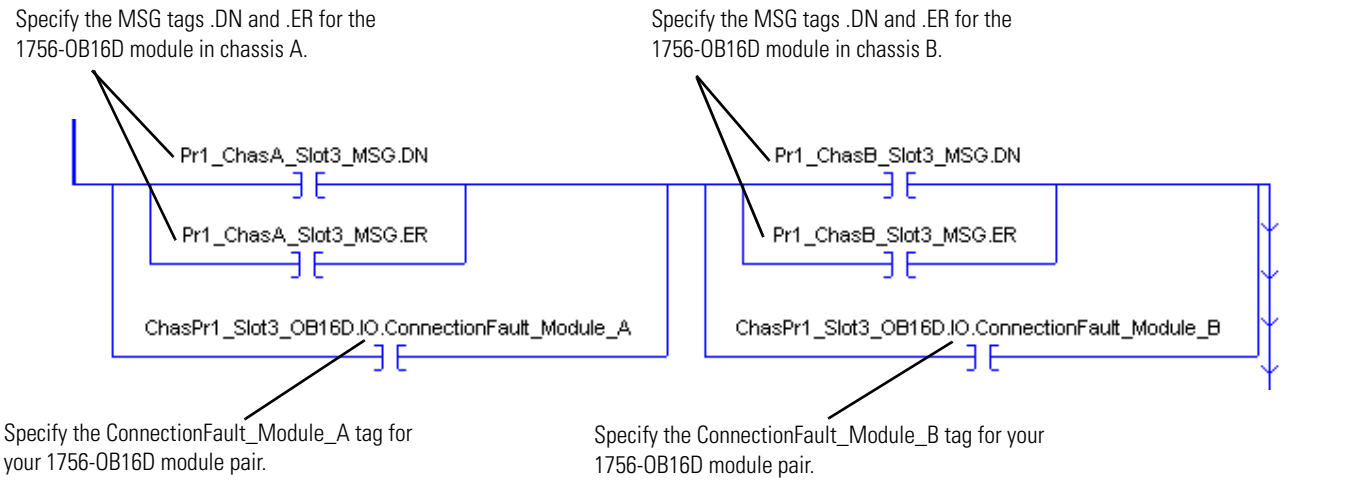
   The programming contained in the first rung initiates the 1756-OB16D module pair's pulse test and moves the data related to the completed pulse test into the 1756-OB16D diagnostic subroutines.

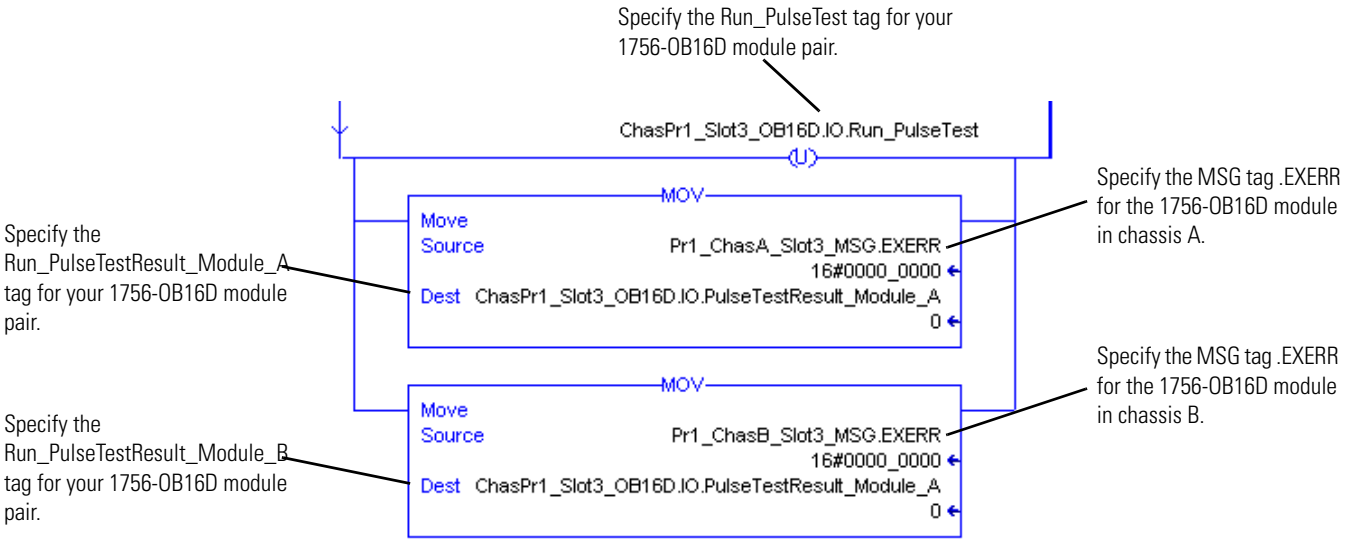
IMPORTANT When specifying OneShot_Bits, **use only OneShot_Bits 2 and 3.**



You edit the MSG instructions contained at the end of this rung during step 3 of this procedure.

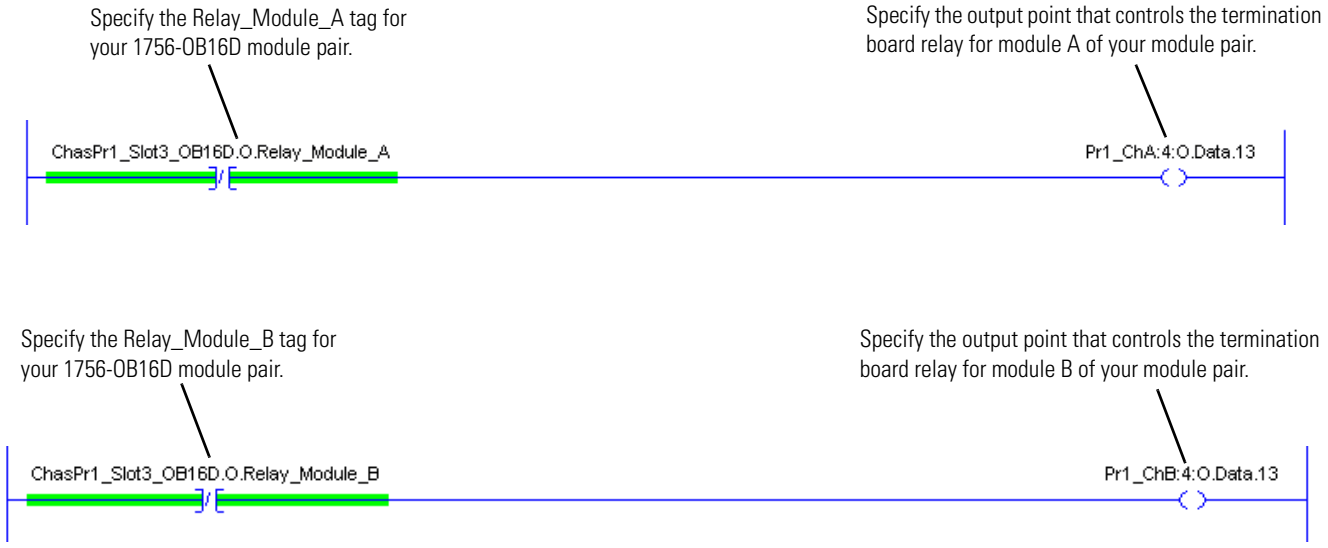
8 / 2011 www.klinkmann.com KLINKMANN


Specify the MSG tags .DN and .ER for the 1756-OB16D module in chassis A.
Specify the MSG tags .DN and .ER for the 1756-OB16D module in chassis B.
Pr1_ChasA_Slot3_MSG.DN
Pr1_ChasA_Slot3_MSG.ER
Pr1_ChasB_Slot3_MSG.DN
Pr1_ChasB_Slot3_MSG.ER
ChasPr1_Slot3_OB16D.IO.ConnectionFault_Module_A
ChasPr1_Slot3_OB16D.IO.ConnectionFault_Module_B
Specify the ConnectionFault_Module_A tag for your 1756-OB16D module pair.
Specify the ConnectionFault_Module_B tag for your 1756-OB16D module pair.



2. In the second and third rungs for the module pair, edit the instruction tags as described in this graphic.

These rungs contain programming that initiates the power disconnect of a faulted 1756-OB16D module.


Specify the Relay_Module_A tag for your 1756-OB16D module pair.
Specify the output point that controls the termination board relay for module A of your module pair.
ChasPr1_Slot3_OB16D.O.Relay_Module_A
Pr1_ChA:4:O.Data.13
Specify the Relay_Module_B tag for your 1756-OB16D module pair.
Specify the output point that controls the termination board relay for module B of your module pair.
ChasPr1_Slot3_OB16D.O.Relay_Module_B
Pr1_ChB:4:O.Data.13

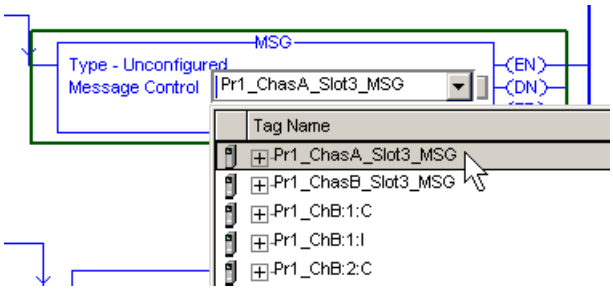
3. In the first rung, edit the MSG instructions to use data specific to your 1756-OB16D module pair.

You must edit each of the two MSG instructions. Edit one MSG instruction to message module A and the other to message module B of the 1756-OB16D module pair.

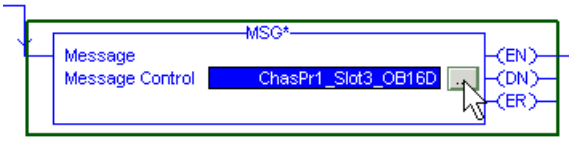
To edit a MSG instruction, complete these steps.

a. Specify the MESSAGE tag you created for the module.

If you need to create MESSAGE tags, see the section titled Adding MESSAGE Tags on page 84.



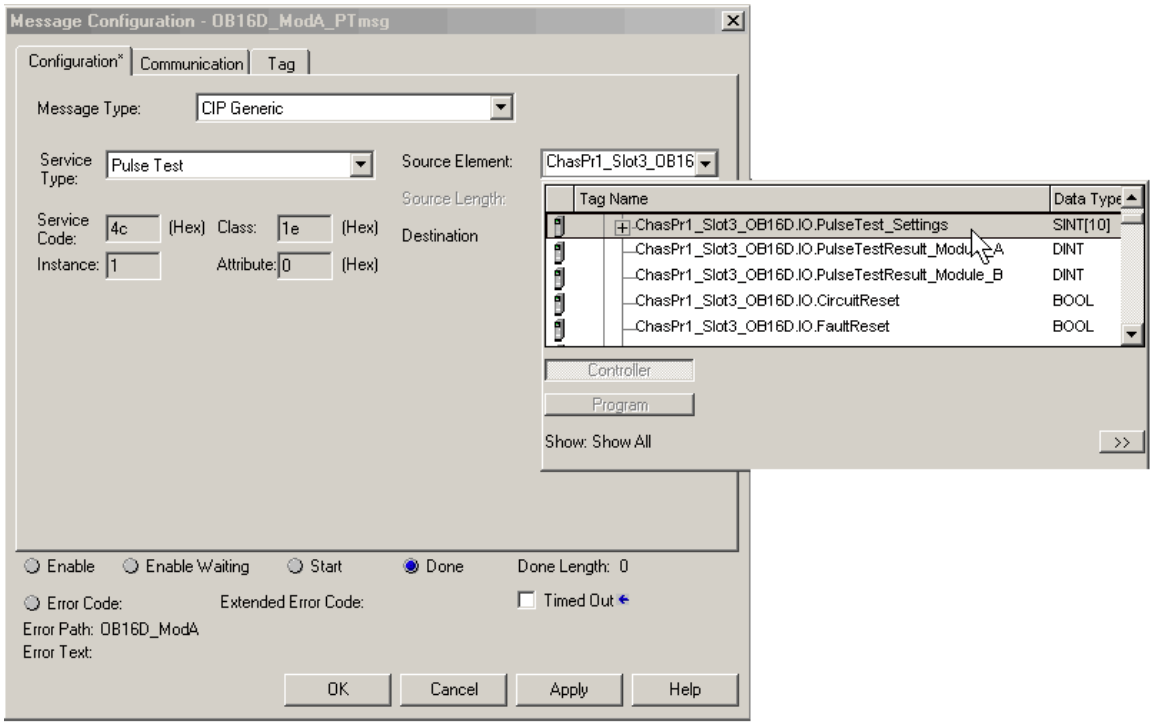
b. Click the View Tag Configuration button located to the right of the Message Control tag.



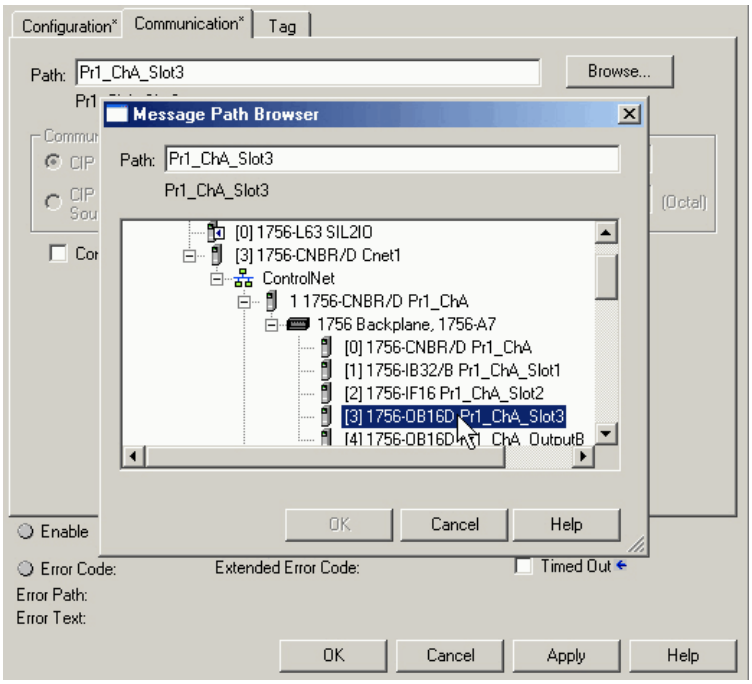
c. In the Configuration tab, specify these properties.

Property	Value
Message Type	CIP Generic
Service Type	Pulse Test
Source Element	PulseTest_Settings (a ModulePair tag)

.



d. In the Communication tab, browse to the 1756-OB16D module.

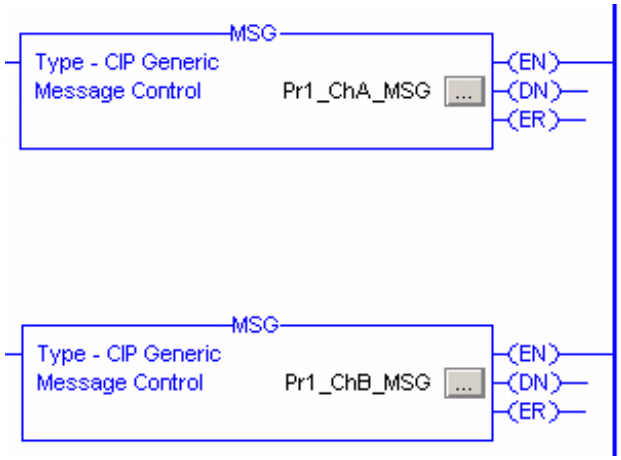


e. Click Apply to accept the changes.

f. Click OK to close the dialog box.

You have completed edits to your MSG instruction.

After you have edited the MSG instructions, they should appear as shown here.



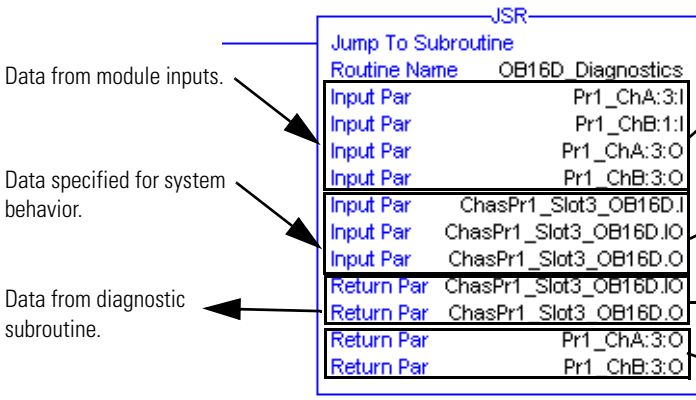
8 / 2011 www.klinkmann.com KLINKMANN

Edit JSR Parameters for the 1756-OB16D Module Pair

The JSR instruction for the 1756-OB16D diagnostic subroutine uses six input parameters and four return parameters. You must edit these parameters so that the tags specific to your system are used.

1756-OB16D Module Pair JSR Parameters

About the Data Used



About the Tags Used

The tags used for these input parameters are system-generated, both input and output (.I and .O) tags that were created when you configured your 1756-OB16D modules.

The tags used for these input parameters are the tags that were generated when you created the ModulePair type tags for the 1756-OB16D module pair.

The diagnostic subroutine returns data to these tags that were generated when you created the ModulePair type tags.

The diagnostic subroutine returns data to these system-generated tags that were created when you configured your 1756-OB16D modules.

Use the following table as a reference when editing your 1756-OB16D JSR parameters.

1756-OB16D Module Pair Tags for Use as JSR Parameters

Parameter	Tag	Description
Input Par	ModuleAName:*X*:I	System-generated input (.I) tags for module A of the pair.
Input Par	ModuleBName:*X*:I	System-generated input (.I) tags for module B of the pair.
Input Par	ModuleAName:X:O	System-generated output (.O) tags for module A of the pair.
Input Par	ModuleBName:X:O	System-generated output (.O) tags for module B of the pair.
Input Par	ModulePairName.I	ModulePair input (I.) tags that contain module pair behavior specification data for both modules of the pair.
Input Par	ModulePairName.IO	ModulePair tags that contain diagnostic status data for both modules of the pair.
Input Par	ModulePairName.O	Tags containing data outputed from the diagnostic subroutine.
Return Par	ModulePairName.IO	ModulePair tags that contain diagnostic status data for both modules of the pair.

1756-OB16D Module Pair Tags for Use as JSR Parameters

Parameter	Tag	Description
Return Par	ModulePairName.0	Tags containing data outputed from the diagnostic subroutine.
Return Par	ModuleAName.0	Data output from the diagnostic subroutine for module A.
Return Par	ModuleBName.0	Data output from the diagnostic subroutine for module B.

You have completed edits to the Call_Code subroutine for a 1756-OB16D module pair. If necessary for your system, repeat steps 1…3 for all of your 1756-OB16D module pairs.

Next Steps

After you have completed the configurations, specifications, and edits described in this chapter, your next step is to program the SIL2 system Main Routine.

See Programming the Fault-tolerant System on page 89 for more information about programming the main routine.

Additional Resources

Resource	Description
Logix5000 Common Programming Procedures Programming Manual, publication 1756-PM001	The programming manual describes common techniques and methods for using RSLogix 5000 software to program Logix5000 controllers.
ControlLogix Controllers User Manual, publication 1756-UM001	This manual explains the general use of ControlLogix controllers.
ControlLogix Redundancy System User Manual, publication 1756-UM523	This user manual explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.
ControlLogix Digital I/O Modules User Manual, publication 1756-UM058	Provides information about digital I/O modules including: features, configuration, and troubleshooting.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Programming the Fault-tolerant System

About This Chapter

This chapter describes suggested methods for programming the fault-tolerant system.

Topic	Page
Programming the Main Routine	105
Basic Input/Output Programming	106
.I and .O Data in Fault-tolerant Programming	106
Example Input/Output Rung	107
Module Pair Fault to Result in System Shutdown	108
Fault Reset Programming	109
Circuit Reset Programming	111
Demand Made Through a 1756-IB32 Module Pair	113
Demand Made Through a 1756-IF16 Module Pair	114
Power-up Sequence	115
Additional Resources	116

Programming the Main Routine

After you have added and configured your JSR instructions and other subroutine elements, you can write the program to control the system in the Main Routine.

This section provides some guidelines and tips for programming the system. It describes some of the many methods you might use to initiate a shutdown of the system in the event of a module pair fault. Also described are some programming methods that might be used to control the system response to a demand on the safety system.

However, these are only guidelines and suggestions as you are responsible for programming the SIL2 system according to your application requirements.

8 / 2011 www.klinkmann.com KLINKMANN

Relationship Between Main Routine and Diagnostic Subroutines

The Main Routine is where you program the system to use data processed and provided by the diagnostic subroutines. While the diagnostic subroutines provide module pair and individual module status data, the program in the Main Routine is what assesses and causes the system response to that data.

Basic Input/Output Programming

Basic input to output programming for I/O modules in the fault-tolerant system varies very little than that for a nonfault-tolerant system. The only difference is in the use of ModulePair tags that appear slightly different than typical system generated tags.

.I and .O Data in Fault-tolerant Programming

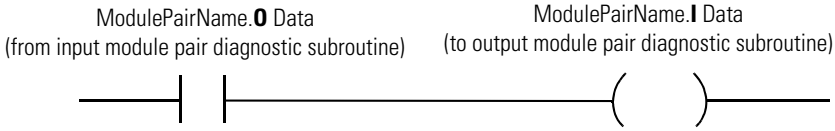
When completing basic input to output programming, remember that the use of module pair tags and the system-generated tags differs because of the .I and .O data designations. For system-generated tags, .I and .O identifies the data's relationship to the module. For ModulePair tags, .I and .O identifies the data's relationship to the diagnostic subroutine.

In nonfault-tolerant programming, a typical input to output rung is programmed as shown.

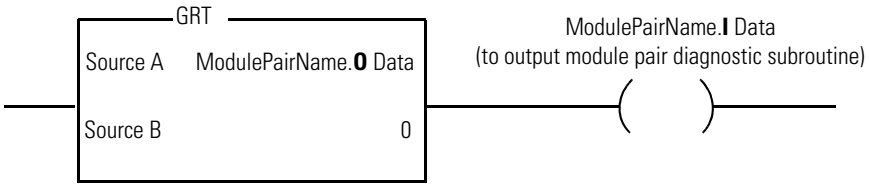
Typical Nonfault-tolerant Input/Output Rung

In fault-tolerant programming, a typical input to output rung is programmed using the ModulePair tags. It appears to be significantly different from the nonfault-tolerant rung because the .I and .O tags are used in reverse order.

Typical Fault-tolerant Digital Input/Output Rung


ModulePairName.O Data
(from input module pair diagnostic subroutine)
ModulePairName.I Data
(to output module pair diagnostic subroutine)

Typical Fault-tolerant Analog Input/Output Rung

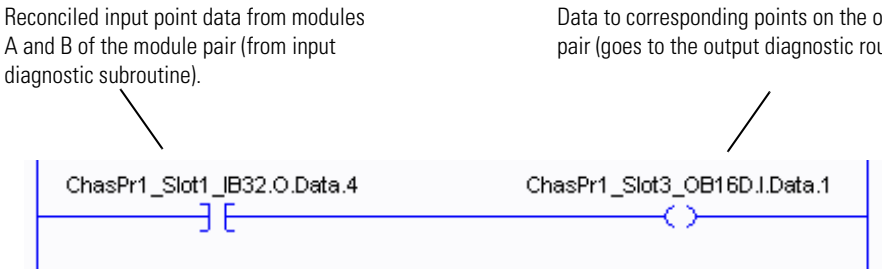


For more information about how data is processed and used in the fault-tolerant program, see Chapter 3, Fault-tolerant Program Elements.

Example Input/Output Rung

This is an example of the basic input/output rung in a fault-tolerant program.

Example of Input/Output Rung

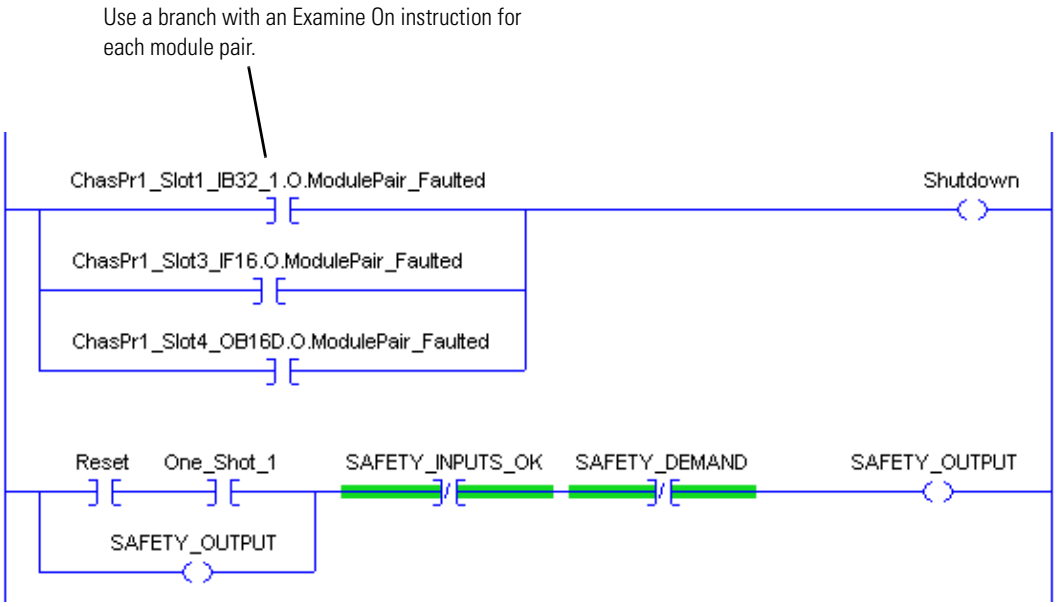

Reconciled input point data from modules A and B of the module pair (from input diagnostic subroutine).
Data to corresponding points on the output module pair (goes to the output diagnostic routine).
ChasPr1_Slot1_IB32.O.Data.4
ChasPr1_Slot3_OB16D.I.Data.1

Module Pair Fault to Result in System Shutdown

Some fault-tolerant applications may require that the system shutdown in the event of a fault at any module pair.

For example, in your application, if both modules of 1756-IB32 module pair is faulted, the resulting safe state for the system may be a total system shutdown.

If your application requires a shutdown when both modules of a module pair are faulted, use programming similar to that shown here.



Fault Reset Programming

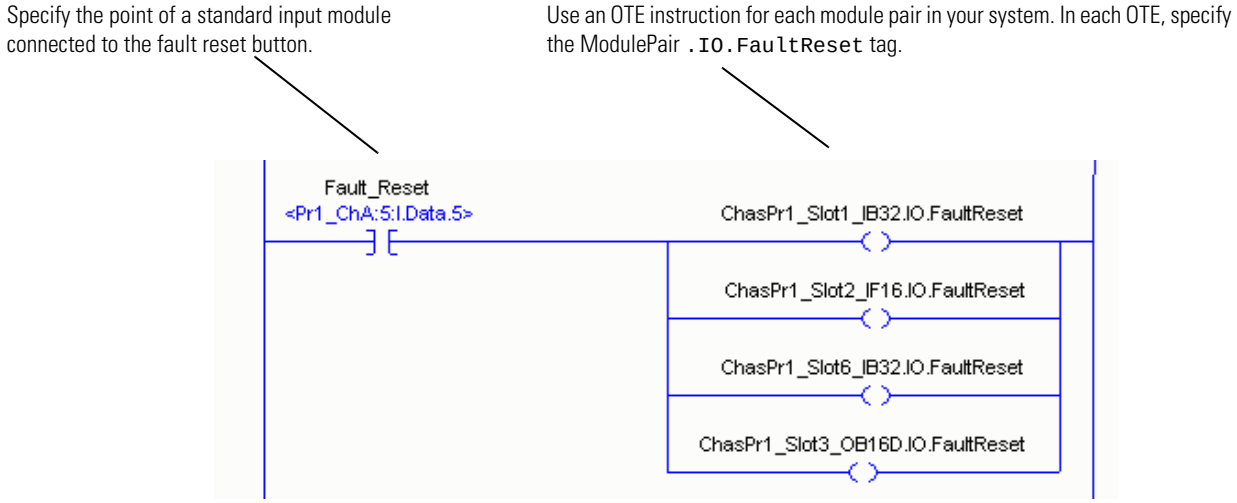
In order to reset ModulePair fault bits in the program after a fault has been corrected, you must use programming to toggle the fault bit (that is, the `IO.FaultReset` tag) for the module pair affected. In many applications, this programming uses an input connected to a pushbutton.

When programming your fault-reset input, these considerations must be made.

- Use an input point that is not a part of the fault-tolerant, module pair inputs (that is, use an input module that is separate from the fault-tolerant system).
- Program the fault reset for each of the module pairs by using an Output Energize (OTE) instruction for each module pair's `.IO.FaultReset` tag.
- You do not need to program the fault reset to be anti-tie down as the programming is already present in the diagnostic subroutines.

Use this example as a reference when programming your fault reset input.

Fault Reset Programming Example


Specify the point of a standard input module connected to the fault reset button.
Use an OTE instruction for each module pair in your system. In each OTE, specify the ModulePair .IO.FaultReset tag.
Fault_Reset
<Pr1_ChA:5:I.Data.5>
ChasPr1_Slot1_IB32.IO.FaultReset
ChasPr1_Slot2_IF16.IO.FaultReset
ChasPr1_Slot6_IB32.IO.FaultReset
ChasPr1_Slot3_OB16D.IO.FaultReset

This programming results in the module status tags being reset to pre-fault values.

When the fault reset bit is toggled, these tag values are reset.

1756-IB32 ModulePair Tags Reset by the IO.FaultReset Bit

- ConnectionFault_Module_A
- ConnectionFault_Module_B
- Chnl_OK_Module_A
- Chnl_OK_Module_B
- ChnlFlt_StuckAtOne_Module_A
- ChnlFlt_StuckAtOne_Module_B
- Module_Pair_Good
- Module_Pair_1oo1
- Module_A_Faulted
- Module_B_Faulted
- Run_1oo1_Countdown

1756-IF16 ModulePair Tags Reset by the IO.FaultReset Bit

- ConnectionFault_Module_A
- ConnectionFault_Module_B
- Chnl_OK_Module_A
- Chnl_OK_Module_B
- ChnlFlt_RefTest_Module_A
- ChnlFlt_RefTest_Module_B
- Module_Pair_Good
- Module_Pair_1oo1
- Module_A_Faulted
- Module_B_Faulted
- Run_1oo1_Countdown

1756-OB16D ModulePair Tags Reset by the IO.FaultReset Bit

- ConnectionFault_Module_A
- ConnectionFault_Module_B
- Chnl_OK_Module_A
- Chnl_OK_Module_B
- ChnlFlt_PulseTest_Module_A
- ChnlFlt_PulseTest_Module_B
- Chnl_Grounded_Module_A
- Chnl_Grounded_Module_B
- Chnl_HWFail_Module_A
- Chnl_HWFail_Module_A
- Chnl_NoLoadOrDCV_Module_A
- Chnl_NoLoadOrDCV_Module_B

Circuit Reset Programming

In the fault-tolerant system, a circuit reset is a manual control used to restart inputs and outputs after a system shutdown has occurred.

When a circuit reset occurs, the data tags for the module pair (that is, the `.I.Data` tags for each module pair) are cleared of the faulted state data and reset to use the sensor data of the modules. This programming restarts the outputs, and therefore the system.

The reset of `.IO.CircuitReset` tag for the 1756-IB32 and 1756-IF16 modules results in `ModulePair.O` data once again reflecting sensor data from the input modules. The reset of `.IO.CircuitReset` for the 1756-OB16D module results in `ModulePair.O` tags once again reflecting the system-requested values of the outputs.

Circuit Reset Programming Considerations

When programming your circuit reset input, these considerations must be made.

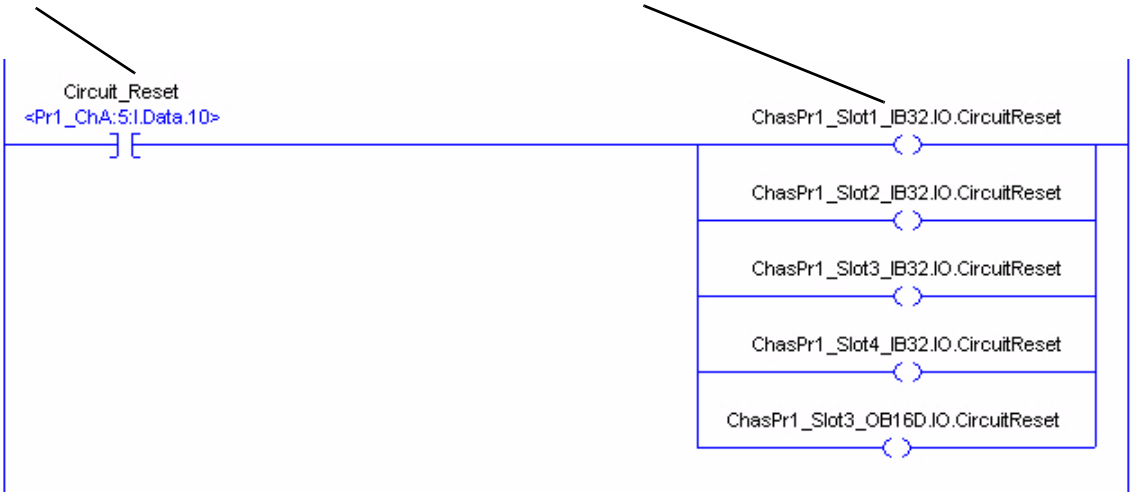
- Use an input point that is not a part of the fault-tolerant, module pair inputs (that is, use an input module that is separate from the fault-tolerant system).
- Program the circuit reset for all of the module pairs by using an Output Energize (OTE) instruction with each ModulePair `.IO.CircuitReset` tag.
- You do not need to program the circuit reset to be anti-tie down as the programming is already present in the diagnostic subroutines.

Use this example as a reference when programming your fault reset input.

Circuit Reset Programming

Specify the point of a standard input module connected to the circuit reset button.

Use an OTE instruction for each module pair in your system. In each OTE, specify the ModulePair `.IO.CircuitReset` tag.



Programming for a Demand on the System

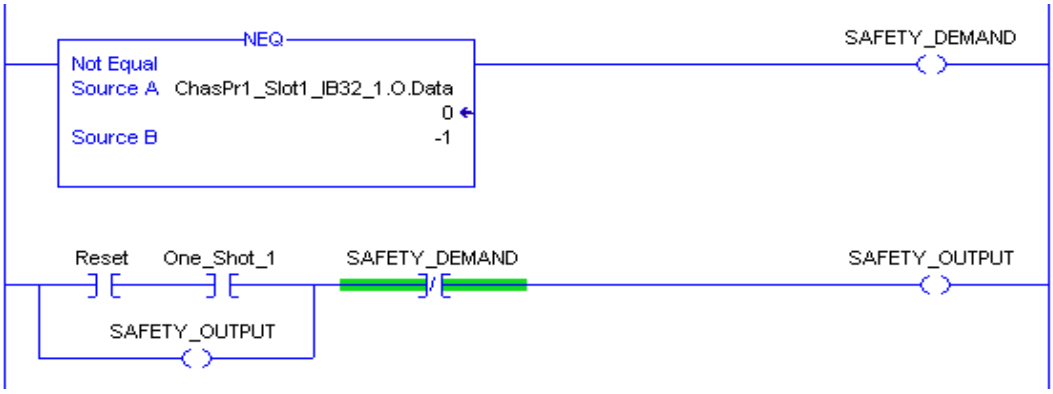
You must also include programming to respond to a demand on the system. These sections provide examples and explanations of programming for a demand on the system.

Demand Made Through a 1756-IB32 Module Pair

This example shows a method of programming for a shutdown when a demand is placed on the system through the 1756-IB32 module pair.

Note that this example is for an 1756-IB32 module pair where all 32 inputs are in use. As it is shown, if any of the digital inputs goes to low (a demand), the system de-energizes.

Example of Demand on the System from an 1756-IB32 Module Pair



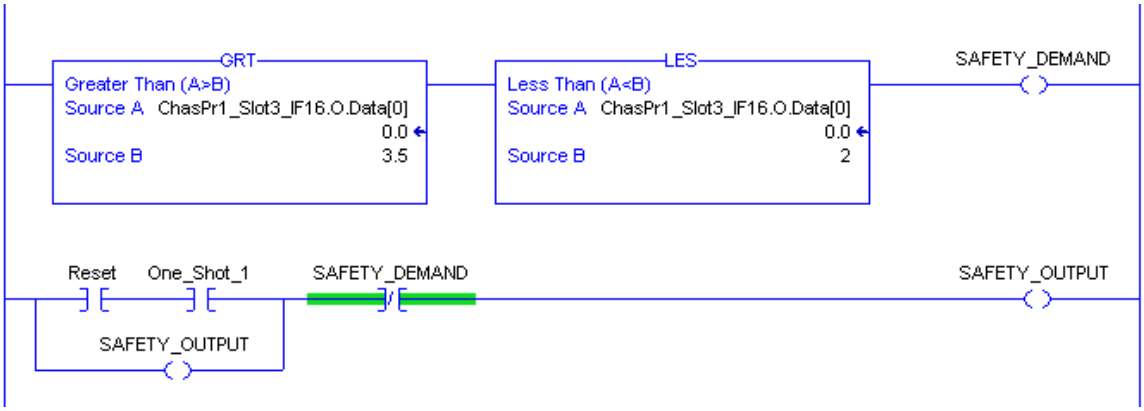
8 / 2011 www.klinkmann.com KLINKMANN

Demand Made Through a 1756-IF16 Module Pair

These examples show methods of programming for a shutdown when a demand is placed on the system through one channel of the 1756-IF16 module pair.

Depending on your application, your programming may use different, but similar, programming than that shown here.

Example of Greater Than and Less Than Instructions to Detect Demand on 1756-IF16 Module Pair



Power-up Sequence

Once you have completed your system programming, you should configure your ControlNet network and download the project to the controller.

After you put the controller into Run mode or you turn on a controller with a fault-tolerant program loaded, there is a sequence of power up steps that you must carry-out. These steps are explained below.

1. Wait five seconds to allow I/O data to be read and established.

IMPORTANT After you have applied power or put the controller into Run mode, the 1756-OB16D module pair faults. This behavior is programmed into the fault-tolerant system in order to protect personnel and machinery from sudden output.

2. Press fault reset to clear the faults of the 1756-OB16D module pair.

   This reset clears the module pair faults and applies power to the 1756-OB*xx* modules).

3. Press circuit reset to set the 1756-OB16D module pair outputs to their commanded state.

4. Press fault reset to carry-out the reference calculations and to verify that all faults of the input modules have been cleared.

After completing these steps, your fault-tolerant system is online and fully operational.

For more information about the fault reset and circuit reset, see these sections:

- Fault Reset Programming, on page 109
- Circuit Reset Programming, on page 111

8 / 2011 www.klinkmann.com KLINKMANN

Additional Resources

Resource	Description
Logix5000 Common Programming Procedures Programming Manual, publication 1756-PM001	The programming manual describes common techniques and methods for using RSLogix 5000 software to program Logix5000 controllers.
ControlLogix Controllers User Manual, publication 756-UM001	This manual explains the general use of ControlLogix controllers.
ControlLogix Redundancy System User Manual, publication 1756-UM523	This user manual explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	This safety reference manual provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

Troubleshooting a Fault-tolerant System

About This Chapter

This chapter explains recommended procedures for troubleshooting a fault-tolerant system. It also contains examples of status information that may result when faults are present in the system.

Topic	Page
Identifying a Faulted Module Pair	118
Identifying a Faulted Module	121
Example of Programming to Identify a Faulted Module Pair	120
Identifying a Faulted Module	121
Replacing a Faulted 1756-IB32 Module	121
1756-IB32 ModulePair Tags to Identify the Type of Module Fault	122
1756-IF16 ModulePair Tags to Identify the Type of Module Fault	123
1756-OB16D ModulePair Tags to Identify the Type of Module Fault	124
Using Resets	125
When to Use the Fault Reset	125
When to Use Circuit Reset	125
Examples of Faults and Resulting Tag Values	126
1756-IF16 Module Pair - Two Modules Faulted	128

Identifying a Faulted Module Pair

In order to identify a faulted module pair, you should examine these tags. Each of these tags is created when you create the ModulePair data type tags for any of the three module types.

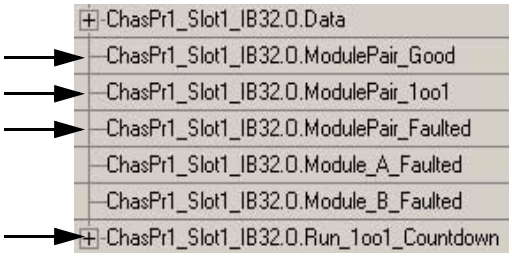
ModulePair Tags Used to Identify a Fault on the Module Pair

Tag	Indicates
O.ModulePair_Good	If both modules of the pair are functioning without faults. 1 = Both modules are functioning properly 0 = A fault is present on one or both modules of the pair
O.ModulePair_1oo1	If the module pair is operating in a 1oo1 configuration (that is, only one module of the pair is functioning properly). 1 = Module pair is operating in a 1oo1 configuration 0 = Both modules are either OK or faulted, and not 1oo1
O.ModulePair_Faulted	If both the modules of the pair are faulted. Depending on your application, a status of 1 at this tag may initiate a shutdown. 1 = Both modules of the pair faulted 0 = Module pair functioning properly or in a 1oo1 configuration.
O.Run_1oo1_Countdown	The time remaining on the TimeToRun1oo1 timer if the module pair is operating in a 1oo1 configuration.

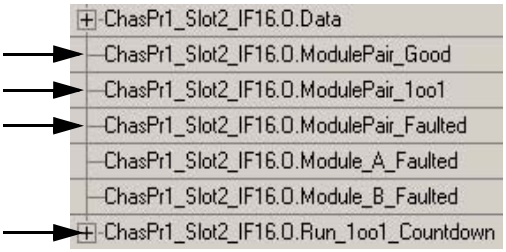
These are the module pair status tags as they appear in the Controller Tags list.

ModulePair Status Tags for Each Module Type

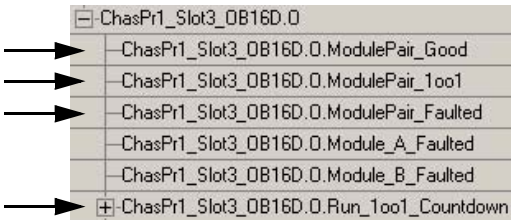
1756-IB32 Module Pair Status Tags



1756-IF16 Module Pair Status Tags



1756-OB16 Module Pair Status Tags



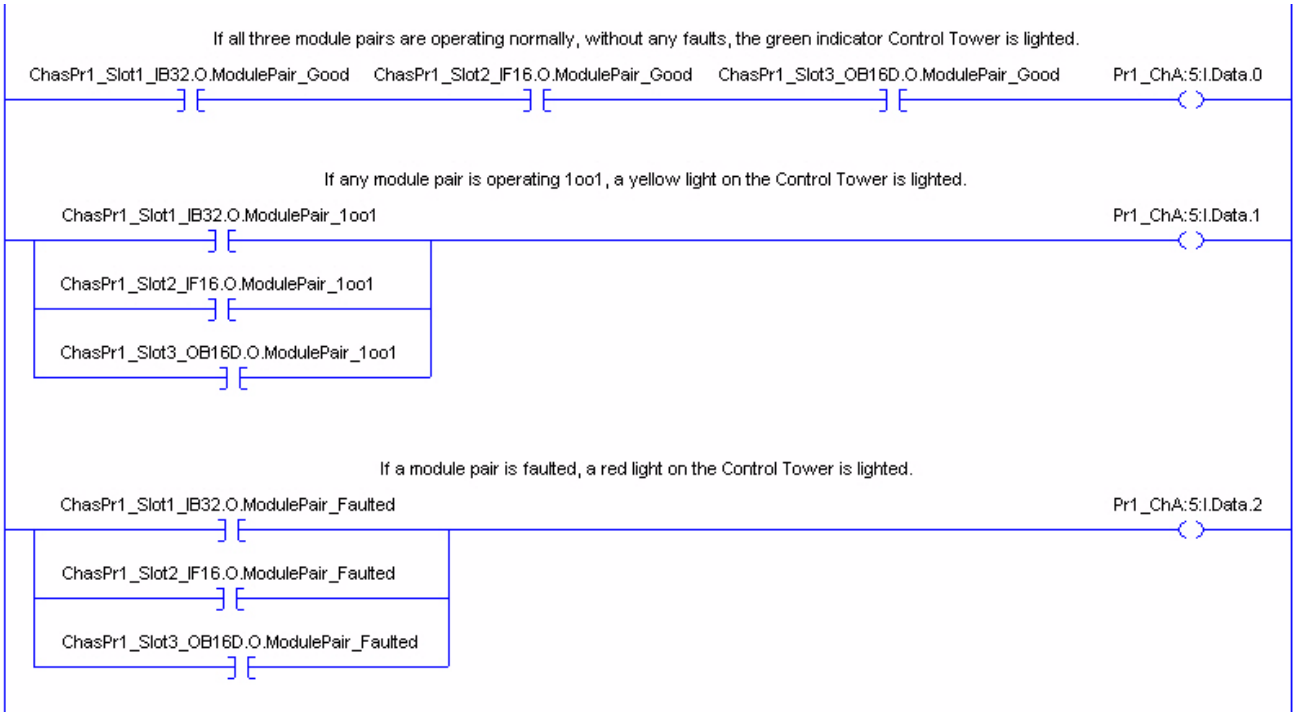
8 / 2011 www.klinkmann.com KLINKMANN

Example of Programming to Identify a Faulted Module Pair

When troubleshooting your fault-tolerant system after a fault on a module pair has occurred, you may choose to examine module status tags by going online with the controller or by programming an HMI or similar notification system to annunciate and identify the faulted module pair.

This example shows one method of programming so that the status of the module pair is displayed. Programming similar to that shown here may be used to demonstrate the status of the module pair on a Control Tower or similar device.

Example of Module Pair Status Programming


If all three module pairs are operating normally, without any faults, the green indicator Control Tower is lighted.
ChasPr1_Slot1_IB32.O.ModulePair_Good
ChasPr1_Slot2_IF16.O.ModulePair_Good
ChasPr1_Slot3_OB16D.O.ModulePair_Good
Pr1_ChA:5:I.Data.0
If any module pair is operating 1oo1, a yellow light on the Control Tower is lighted.
ChasPr1_Slot1_IB32.O.ModulePair_1oo1
Pr1_ChA:5:I.Data.1
ChasPr1_Slot2_IF16.O.ModulePair_1oo1
ChasPr1_Slot3_OB16D.O.ModulePair_1oo1
If a module pair is faulted, a red light on the Control Tower is lighted.
ChasPr1_Slot1_IB32.O.ModulePair_Faulted
Pr1_ChA:5:I.Data.2
ChasPr1_Slot2_IF16.O.ModulePair_Faulted
ChasPr1_Slot3_OB16D.O.ModulePair_Faulted

Identifying a Faulted Module

In order to identify a faulted module, you should examine these tags. Each of these tags is created when you create the ModulePair data type tags for any of the three module types.

ModulePair Tags Used to Identify a Faulted Module

Tag	Indicates
O.Module_A_Faulted	The fault status of module A. 1 = Module A faulted 0 = Module A functioning properly
O.Module_B_Faulted	The fault status of module B. 1 = Module B faulted 0 = Module B functioning properly

Once you have used the tags listed above to identify a faulted module, there are additional tags you can view to determine what type of fault exists on the module.

Each module type uses different tags to identify the type of fault. Use the section specific to your module to determine which type of fault exists on the module.

Replacing a Faulted 1756-IB32 Module

If your 1756-IB32 module pair is operating 1oo1 at a point-level (that is one module of the pair has a faulted point and the other module is fully-functional), removing the swing-arm of the module with 1…31 faulted points causes your system to fail-to-safe due to a miscompare.

The miscompare occurs because data from the unfaulted points of the module continue to be used and checked by the diagnostic subroutine. Removing the swing-arm results in the remaining unfaulted points going low (0) and a miscompare of data occurs.

IMPORTANT To avoid a shutdown due to a miscompare, remove the entire 1756-IB32 module from the chassis before removing the swing-arm.

8 / 2011 www.klinkmann.com KLINKMANN

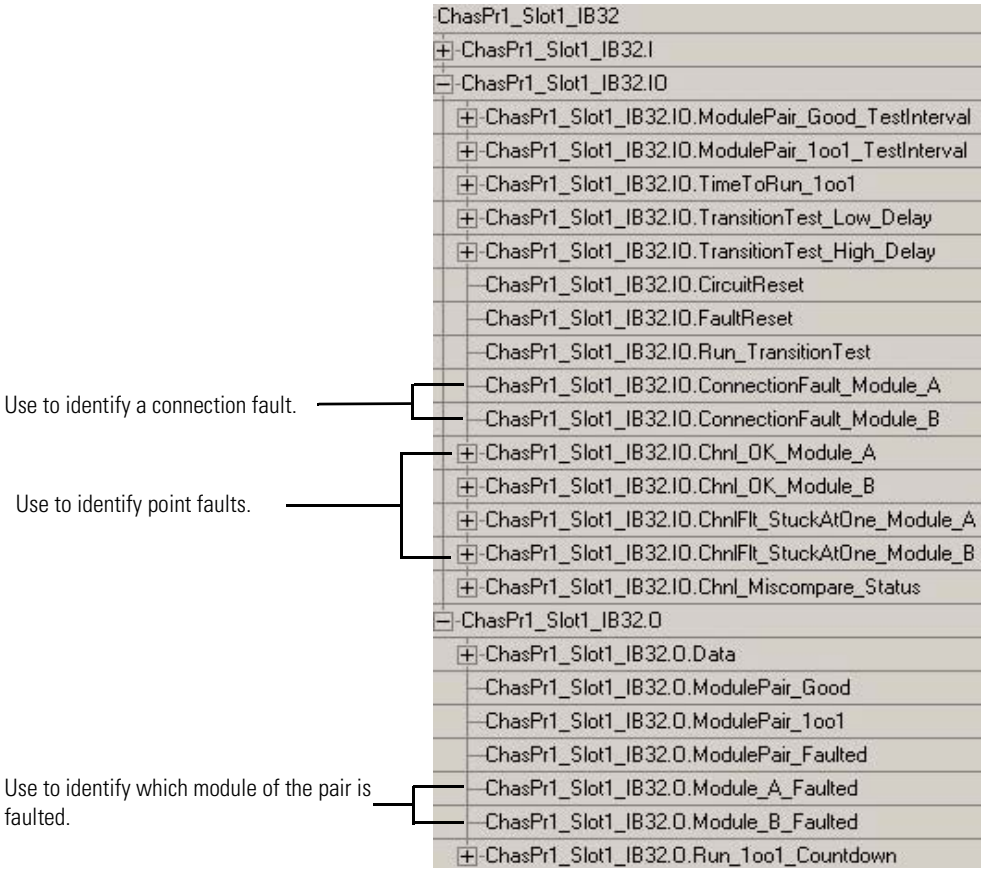
1756-IB32 ModulePair Tags to Identify the Type of Module Fault

The ModulePair data type for the 1756-IB32 module provides tags that can help identify these types of faults:

- Connection and communication faults.
- Points on the module faulted (for example, a miscompare or stuck-at-one condition).
- Point or points fail to transition from one to zero during transition test (for example, due to an internal short).

These are the tags that contain the 1756-IB32 module status data and can be used to determine the type of module fault.

1756-IB32 Module Status Tags



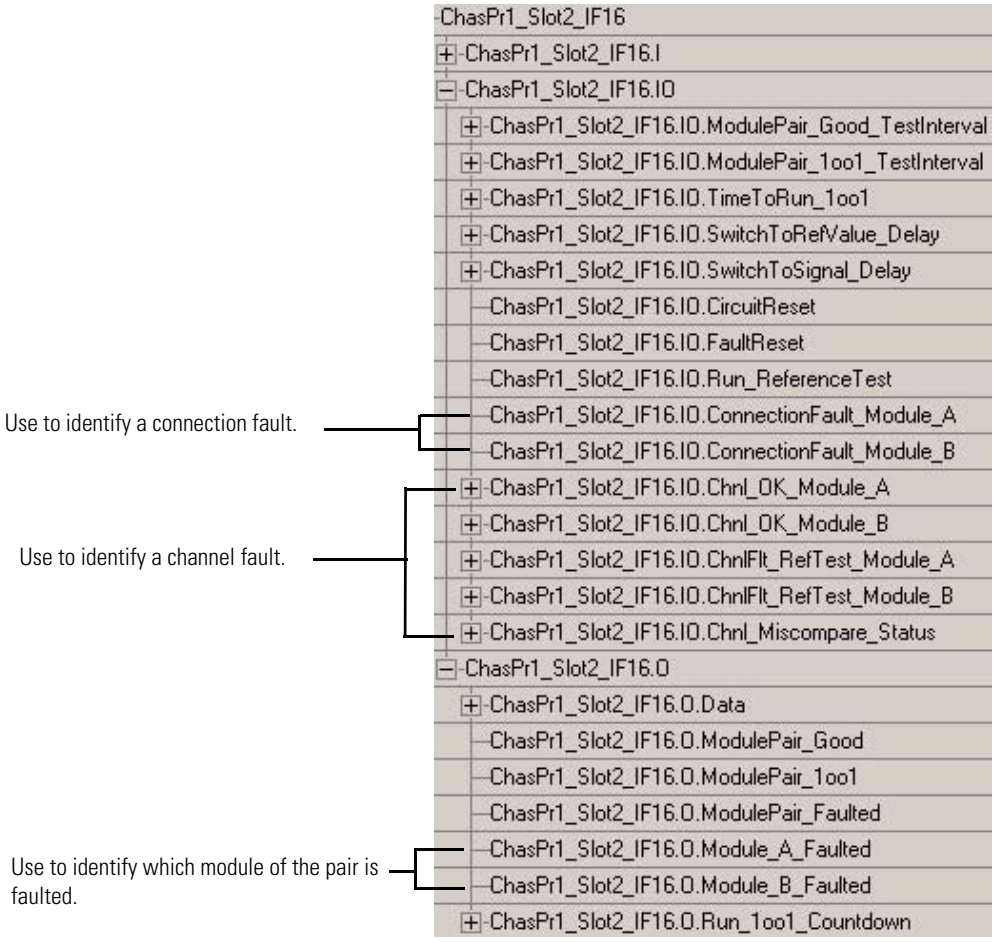
1756-IF16 ModulePair Tags to Identify the Type of Module Fault

The ModulePair data type for the 1756-IF16 module provides tags that can help identify these types of faults:

- Connection and communication faults.
- Channels on the module faulted (for example, due to a miscompare or over/under range).
- Channels faulted as determined during the reference test.

These are the tags that contain the 1756-IF16 module status data and can be used to determine the type of module fault.

1756-IF16 Module Status Tags



8 / 2011 www.klinkmann.com KLINKMANN

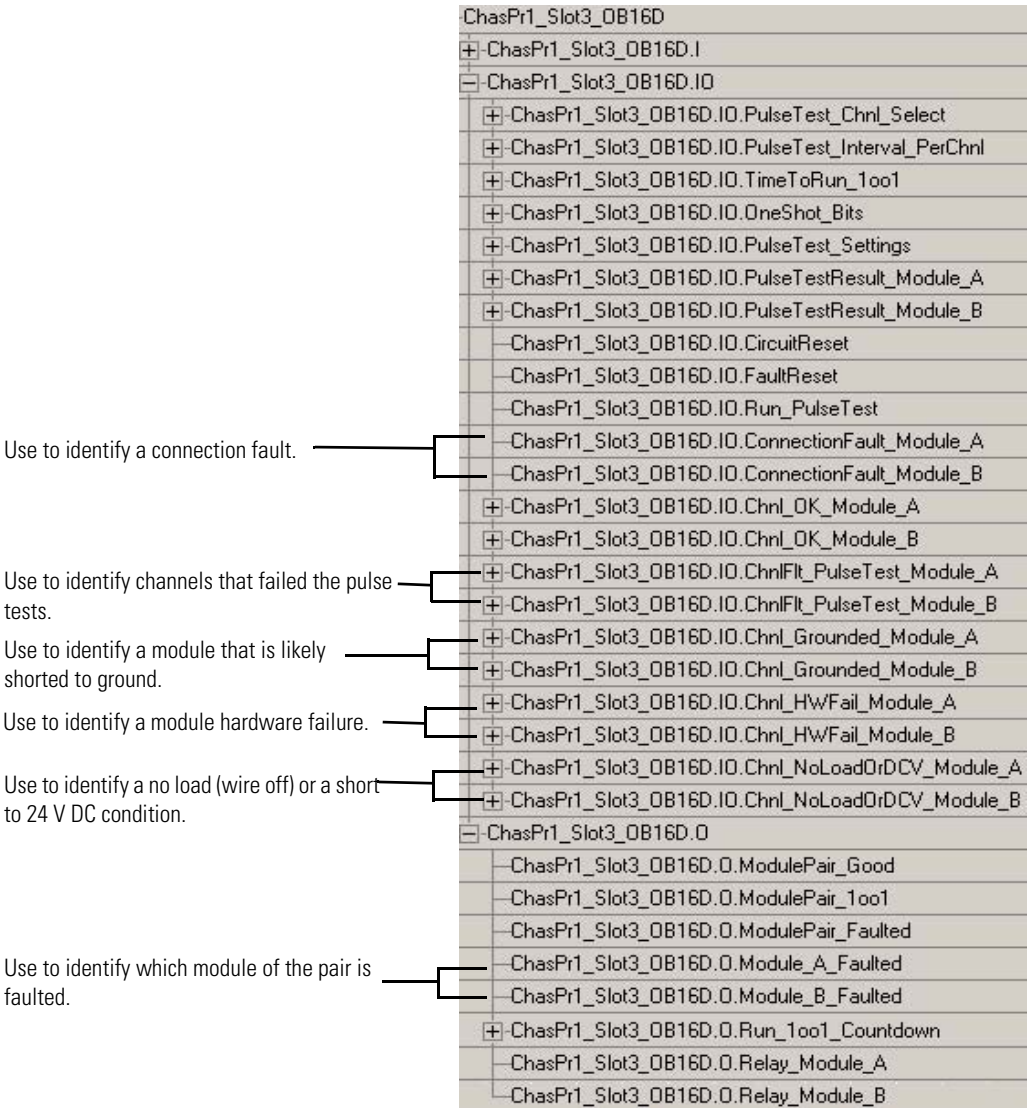
1756-OB16D ModulePair Tags to Identify the Type of Module Fault

The ModulePair data type for the 1756-OB16D module provides tags that can help identify these types of faults:

- Connection and communication faults.
- No load conditions (detects no load conditions only between the output module and termination board).
- Points stuck at low.
- Points stuck at high.
- Other hardware failures.

These are the tags that contain the 1756-OB16D module status data and can be used to determine the type of module fault.

1756-OB16D Module Status Tags



Using Resets

After you have finished troubleshooting and repairing a faulted module condition, you must reset the system so that the faults are cleared and the system operates using the data from the repaired module.

Depending on the type of fault and the configuration the system is running in, you may be required to reset both the fault status tags and the data tags (by using the circuit reset).

When to Use the Fault Reset

After you have repaired or replaced the faulted module, or corrected any other issues that might cause a module fault, you must use the Fault Reset button.

If you program the Fault Reset button as instructed in Chapter 5, in the section titled Fault Reset Programming (page 109), pressing the fault reset button results in all of the module fault status tags being reset. However, module data tags are not reset.

If your system was operating in a 1oo1 configuration at the module fault, the fault reset is the only action you need to take in order to enable the system to use data from the newly-repaired module.

When to Use Circuit Reset

If both modules of the pair are faulted, you must use the circuit reset after using the fault reset.

Because the fault reset clears only the module fault status tags, the faulted values are still present in the module data tags. 1756-IB32 module data tags fault values are 0, and 1756-IF16 fault values are those specified in the ModulePair tags `ChnlValues_at_Fault`.

Using the circuit reset, (if programmed as described in Chapter 5, in the section titled Circuit Reset Programming, on page 111) the faulted data values are cleared and the system uses the sensor data from the modules.

Examples of Faults and Resulting Tag Values

These examples show how the ModulePair tags appear before and after a certain module fault occurs. Each column of the tables indicates what action has taken place. The tags listed in the rows of the columns indicate the tag values after the action has occurred.

1756-IB32 Module Pair - One Module Faulted

In this example, module A of the 1756-IB32 module pair has a stuck-at-one condition caused by an internal short. The stuck-at-one condition is detected during the next transition test.

This table shows which tags values change from the time the transition test detects the fault to the point when the fault is cleared and the system is operating using data from the repaired module.

Tag Values After a Stuck-At-One Condition Detected on a 1756-IB32 Module

Tag	Values During Normal Operation (No Faults)	Values After Fault Detected	Values After Faults Repaired and Fault Reset	Values After Circuit Reset
ConnectionFault_Module_A	0	0	0	N/A(1)
ConnectionFault_Module_B	0	0	0	N/A(1)
Chnl_OK_Module_A	1 (at each point)	0 (at affected points)	1 (at each point)	N/A(1)
Chnl_OK_Module_B	1 (at each point)	1 (at each point affected)	1 (at each point)	N/A(1)
Chnl_Miscompare_Status	0 (at each point)	0 (at each point)	0 (at each point)	N/A(1)
ChnlFlt_StuckAtOne_Module_A	0	1 (at each point affected)	0	N/A(1)
ChnlFlt_StuckAtOne_Module_B	0	0	0	N/A(1)
Data	From modules A and B	From module B	From modules A and B	N/A(1)
ModulePair_Good	1	0	1	N/A(1)
Module_Pair_1oo1	0	1	0	N/A(1)
ModulePair_Faulted	0	0	0	N/A(1)
Module_A_Faulted	0	1	0	N/A(1)
Module_B_Faulted	0	0	0	N/A(1)
Run_1oo1_Countdown	Preset	Counting down	Preset	N/A(1)

(1) Circuit reset is not needed in this case because the system did not stop using data from the module pair.

1756-IF16 Module Pair - One Module Faulted and Removed

In this example, module B of the 1756-IF16 module pair has a fault caused by an internal short. The tag value changes are shown after the fault is identified by the reference test, when the module is removed for repair, and after the module has been replaced and the faults reset.

Tag Values After Faulted Channel Detected on a 1756-IF16 Module

Tags	Values During Normal Operation (No Faults)	Values After Fault Detected	Values After Module B Removed	Values After Module B Replaced and Fault Reset
ConnectionFault_Module_A	0	0	0	0
ConnectionFault_Module_B	0	0	1	0
Chnl_OK_Module_A	1 (at each channel)	1 (at each channel)	1 (at each channel)	1 (at each channel)
Chnl_OK_Module_B	1 (at each channel)	0 (at affected channel)	0 (at each channel)	1 (at each channel)
ChnlFlt_RefTest_Module_A	0	0 (at each channel)	0 (at each channel)	0 (at each channel)
ChnlFlt_RefTest_Module_B	0	1 (at affected channels)	0 (at each channel)	0 (at each channel)
Chnl_Miscompare_Status	0	0 (at each channel)	0 (at each channel)	0 (at each channel)
Data	From modules A and B	From module A	From module A	From modules A and B
ModulePair_Good	1	0	0	1
Module_Pair_1oo1	0	1	1	0
ModulePair_Faulted	0	0	0	0
Module_A_Faulted	0	0	0	0
Module_B_Faulted	0	1	1	0
Run_1oo1_Countdown	Preset	Counting down	Counting down	Preset

1756-IF16 Module Pair - Two Modules Faulted

In this example, a fault occurs on module B of the module pair. Then, while operating 1oo1, module A faults as well. The table shows the progression of tag values through the initial fault on module B through the circuit reset.

Tag Values After 1756-IF16 Module Pair Faulted

Tags	Values During Normal Operation (No Faults)	Values After Module B Fault Detected	Values After Module A Fault Detected	Values After Faults Corrected and Fault Reset	Values After Circuit Reset
ConnectionFault_Module_A	0	0	0	0	0
ConnectionFault_Module_B	0	0	0	0	0
Chnl_OK_Module_A	1 (at each channel)	1 (at each channel)	0 (at affected channels)	1 (at each channel)	1 (at each channel)
Chnl_OK_Module_B	1 (at each channel)	0 (at affected channels)	0 (at affected channels)	1 (at each channel)	1 (at each channel)
ChnlFlt_RefTest_Module_A	0 (at each channel)	0 (at each channel)	1 (at affected channels)	0 (at each channel)	0 (at each channel)
ChnlFlt_RefTest_Module_B	0 (at each channel)	1 (at affected channels)	1 (at affected channels)	0 (at each channel)	0 (at each channel)
Chnl_Miscompare_Status	0 (at each channel)	0 (at each channel)	0 (at each channel)	0 (at each channel)	0 (at each channel)
Data	From modules A and B	From module A	As set for fault values	As set for fault values	From modules A and B
ModulePair_Good	1	0	0	1	1
Module_Pair_1oo1	0	1	0	0	0
ModulePair_Faulted	0	0	1	0	0
Module_A_Faulted	0	0	1	0	0
Module_B_Faulted	0	1	1	0	0
Run_1oo1_Countdown	Preset	Counting down	Preset	Preset	Preset

Additional Resources

Resource	Description
ControlLogix Digital I/O Modules User Manual, publication 1756-UM058	Provides information about digital I/O modules including: features, configuration, and troubleshooting.
Logix5000 Common Programming Procedures Programming Manual, publication 1756-PM001	The programming manual describes common techniques and methods for using RSLogix 5000 software to program Logix5000 controllers.
ControlLogix Controllers User Manual, publication 1756-UM001	Explains the general use of ControlLogix controllers.
ControlLogix Redundancy System User Manual, publication 1756-UM523	Explains how to design, install, configure, and troubleshoot a redundant ControlLogix system.
Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001	Provides information regarding ControlLogix components for use in SIL2 applications. Topics include hardware, software, and programming components.

You can view or download Rockwell Automation publications at http://literature.rockwellautomation.com. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

8 / 2011
www.klinkmann.com KLINKMANN

Notes:

SIL2 Remote I/O Fault-tolerance Tags

About This Appendix

This appendix provides tag names, purposes, and values for each type of I/O module available for use in the ControlLogix SIL2 fault-tolerant system. Use this appendix as a reference when programming your SIL2 fault-tolerant system.

Topic	Page
1756-IB32 ModulePair Tags	131
1756-IB32 ModulePair Tags for System Behavior	131
1756-IB32 Module Status Tags	133
1756-IB32 ModulePair Tags for Use in Programming	135
1756-IB32 Hidden Tags, Not for Use	136
1756-IF16 ModulePair Tags	137
1756-IF16 ModulePair Tags for System Behavior	137
1756-IF16 Module Status Tags	138
1756-IF16 ModulePair Tags for Use in Programming	141
1756-IF16 Hidden Tags, Not for Use	142
1756-OB16D Module Pair Tags	143
1756-OB16D ModulePair Tags for System Behavior	143
1756-OB16D Module Status Tags	144
1756-OB16D ModulePair Tags for Use in Programming	146
1756-OB16D Hidden Tags, Not for Use	147

1756-IB32 ModulePair Tags

The tags provided in the following tables are used to configure, specify, and monitor 1756-IB32, DC input module behavior in a ControlLogix fault-tolerant system.

1756-IB32 ModulePair Tags for System Behavior

You must enter values for each these 1756-IB32 ModulePair tags. For some tags, the value specified is required. For others, the values are recommended.

1756-IB32 ModulePair Tags Used to Specify System Behavior

Tag Name	Description	Value	Required or Recommended
I.Safety_Input_Select	Use to select or deselect the inputs that are used for safety functions.	1 (at each point)	Required
I.Miscompare_Test_Limit	Defines the number of times a miscompare between points is permitted before a fault is declared.	4(1)	Recommended
IO.ModulePair_Good_TestInterval	Time, in ms, between transition tests. The program uses this value when the module pair is without faults.	86400000 (24 hours)	Recommended
IO.ModulePair_1oo1_TestInterval	Time, in ms, between transition tests if the module pair is operating in a 1oo1 configuration. The program uses this value when a fault is present on one module of the pair.	3600000 (1 hour)	Recommended
IO.TimeToRun_1oo1.PRE	User-defined time, in ms, for the 1oo1 countdown timer that is the repair time.	28800000 (8 hours)	Recommended
IO.TransitionTest_Low_Delay.PRE	Amount of time, in ms, delayed to allow the inputs to transition from high to low before checking the results of the transition test. The amount of time to delay should be determined by adding your program scan time to the NUT. For example, if your total program scan time is 80 ms and your NUT is 20 ms, you should set your TransitionTest_Low_Delay value to 100 ms.	100(2)	Recommended
IO.TransitionTest_High_Delay.PRE	Amount of time, in ms, delayed to allow inputs to transition to high before normal operation is resumed after a transition test. The amount of time to delay should be determined by adding your program scan time to the NUT. For example, if your total program scan time is 80 ms and your NUT is 20 ms, you should set your TransitionTest_Low_Delay value to 100 ms.	100(2)	Recommended

(1) The value of four is strongly recommended in order to avoid nuisance trips as well as to provide a timely safety response. If you choose to specify a value lower than four, your system may experience nuisance trips. However, you may choose to lower the value in order to decrease amount of time between a fault and the system response. Setting a value larger then four is not recommended as the response to a fault may be too long for most safety applications.

(2) When specifying your TransitionTest_Low_Delay and TransitionTest_High_Delay values, remember that the system is functioning on the last-known verified data during these periods. If an input connected to the module pair changes (for example, if an E-stop is pressed), it will not be processed until the total time of these two values has expired and the system has stopped using the last-known verified data

1756-IB32 Module Status Tags

The module status tags provide diagnostic information for the module pair. These tags are used in several ways in the fault-tolerant system. Uses include:

- in the main routine to determine system behavior.
- in the subroutine to determine and report module pair status.
- in conjunction with HMI and other indicators of system status.

1756-IB32 Module Status Tags

Tag Name	Description
IO.ConnectionFault_Module_A	Indicates the status of the connection to module A. 1 = Connection lost 0 = Connection good
IO.ConnectionFault_Module_B	Indicates the status of the connection to module B. 1 = Connection lost 0 = Connection good
IO.Chnl_OK_Module_A	Bit-level indicators of what points are operating without fault on module A. 1 = Point is functional 0 = Point is faulted
IO.Chnl_OK_Module_B	Bit-level indicators of what points are operating without fault on module B. 1 = Point is functional 0 = Point is faulted
IO.ChnlFlt_StuckAtOne_Module_A	Bit-level indicators of points on module A that are stuck at one after the transition test. 1 = Point is stuck at one 0 = Point is functional
IO.ChnlFlt_StuckAtOne_Module_B	Bit-level indicators of points on module B that are stuck at one after the transition test. 1 = Point is stuck at one 0 = Point is functional
IO.Chnl_Miscompare_Status	Bit-level indicators that show what points of the module pair do not match each other (miscompare). 1 = Point status between modules is different 0 = Point status is the same
O.ModulePair_Good	Status bit that indicates that both modules of the module pair are functioning properly. 1 = Module pair functioning properly 0 = Fault present (on one or both modules)

1756-IB32 Module Status Tags

Tag Name	Description
O.ModulePair_1oo1	Status bit that indicates the module pair is operating 1oo1. 1 = Operating 1oo1 0 = Either both modules of pair are OK or are faulted (that is, not in 1oo1 operation)
O.ModulePair_Faulted	Status bit indicates that both modules of the module pair have at least one fault. The system has failed to safe. 1 = Both modules of pair faulted 0 = Both modules of pair OK
O.Module_A_Faulted	Status bit indicates that module A of the pair has at least one fault. 1 = Module A faulted 0 = Module A OK
O.Module_B_Faulted	Status Bit indicating that module B of the module pair has at least one fault. 1 = Module B faulted 0 = Module B OK
O.Run_1oo1_Countdown	Indicates the time remaining on the 1oo1 countdown timer. The value is determined using the TimeToRun_1oo1tag value and is shown in seconds.

1756-IB32 ModulePair Tags for Use in Programming

These tags are to be used in either the main routine or in call code programs. Your program uses the data in these tags to determine system behavior.

For example, your call code routine should examine the `Run_TransitionTest` tag. If the value of this tag is at 1, a transition test is run on the module pair.

1756-IB32 Tags for Use in Programming

Tag Name	Description
0.Data	During normal operation these input bits are the reconciled values of two points on the module pair. During 1oo1 operation, these input bits contain data from the unfaulted module of the pair.
I0.CircuitReset	Using programming in the Main Routine, this bit is set manually and clears the 0 value from the data tags and causes the sensor values from the input modules to be used after a fault or demand on the system.
I0.FaultReset	Using programming in the Main Routine, this bit is set manually and resets the module status tags after a fault or demand on the system.
I0.Run_TransitionTest	Used in the IB32_Subroutine_Call_Code, this tag value is a precondition for the DC output that controls the relay on the module pair's termination board.

1756-IB32 Hidden Tags, Not for Use

Similar to the inability to access the diagnostic subroutines, there are tags within the program provided by Rockwell Automation that cannot be accessed or altered.

You cannot see these tags, however, in order to avoid potential conflicts within the program, you should not create tags with the same names.

When creating tags for your application, do not use these tags names.

- DataCompareCounter
- L_Scr_a
- QualityMask1
- QualityMask2
- OneShot_Bits
- TransitionTestInterval
- FaultResetTimer
- Fault
- Data
- Good2Go

1756-IF16 ModulePair Tags

The tags provided in the following tables are used to configure, specify, and monitor 1756-IF16 analog input module behavior in a ControlLogix fault-tolerant system.

1756-IF16 ModulePair Tags for System Behavior

You must enter values for each these 1756-IF16 ModulePair tags. For some tags, the value specified is required. For others, the values are recommended.

1756-IF16 ModulePair Tags Used to Specify System Behavior

Tag Name	Description	Value	Required or Recommended
I.Safety_Input_Select	Enter 1 for any analog input channel being used.(2)	1 at each channel used 0 at each unused channel	Required
I.ChnlCompare_Deadband(1)	Specifies the +/- deadband when the data from two inputs is compared. Entered in percentage of engineering units.	0.05 (at each channel), that is 5%	Recommended
I.ReferenceTest_Deadband(1)	Specifies the +/- deadband between the reference voltage and actual value when a reference test takes place. Entered in percentage of engineering units.	0.05 (at each channel), that is 5%	Recommended
I.ChnlValues_at_Fault[16]	Sets the channel values to be used in the event of a faulted module pair. These values should be entered in engineering units.	0	Recommended
I.Miscompare_Test_Limit	Defines the number of times a miscompare between channels is permitted before a fault is declared.	4(3)	Recommended
IO.ModulePair_Good_TestInterval.PRE	Time, in ms, between transition tests. The program uses this value when the module pair is without faults.	86400000 (24 hours)	Recommended
IO.ModulePair_1oo1_TestInterval.PRE	Time, in ms, between Transition Tests if the module pair is operating in a 1oo1 configuration. The program uses this value when a fault is present on one module of the pair.	3600000 (1 hour)	Recommended
IO.TimeToRun_1oo1.PRE	User-defined time, in ms, for the 1oo1 countdown timer that is the repair time.	28800000 (8 hours)	Recommended

1756-IF16 ModulePair Tags Used to Specify System Behavior

Tag Name	Description	Value	Required or Recommended
IO.SwitchToRefValue_Delay.PRE	Amount of time, in ms, delayed to allow the inputs to transition to the reference values before checking the results of the reference test. This value should be equal or greater than your analog module pair's RTS rate.	500(4)	Recommended
IO.SwitchToSignal_Delay.PRE	Amount of time, in ms, delayed to allow the inputs to transition to the field signal values before normal operation is resumed. This value should be equal or greater than your analog module pair's RTS rate.	500(4)	Recommended

(1) If changes are made to the ChnlCompare_Deadband or to the ReferenceTest_Deadband tag values after the initial fault-tolerant program is downloaded to and running on the controller, then you must press fault-reset so that the IF16_RefCal subroutine is carried out and the new deadband values are implemented. The changes to these tags are not implemented into the program until the IF16_RefCal subroutine is run.

(2) Unused safety input channels cannot be used for any other purposes (that is, they cannot be used as nonfault-tolerant I/O channels). We recommend that you configure unused channels for voltages of 0…5V and then jumper or ground unused channels to keep channel values within range.

(3) The value of four is strongly recommended in order to avoid nuisance trips as well as to provide a timely safety response. If you choose to specify a value lower than four, your system may experience nuisance trips. However, you may choose to lower the value in order to decrease amount of time between a fault and the system response. Setting a value larger then four is not recommended as the response to a fault may be too long for most safety applications.

(4) When specifying your SwitchToRefValue_Delay and SwitchToSignal_Delay values, remember that the system is functioning on the last-known verified data during these periods. If an input connected to the module pair changes, it will not be processed until the total time of these two values has expired and the system has stopped using the last-known verified data.

1756-IF16 Module Status Tags

The module status tags are used in several ways. Uses include:

- in the main routine to determine system behavior.
- in the subroutine to detemine and report module pair status.
- in conjunction with HMI and other indicators of system status.

1756-IF16 Module Status Tags

Tag Name	Description
ConnectionFault_Module_A	Indicates the status of the connection to module A. 1 = Connection lost 0 = Connection good
ConnectionFault_Module_B	Indicates the status of the connection to module B. 1 = Connection lost 0 = Connection good
Chnl_OK_Module_A	Bit-level indicators of what channels are operating without fault on module A. 1 = Channel is functional 0 = Channel is faulted
Chnl_OK_Module_B	Bit-level indicators of what channels are operating without fault on module B. 1 = Channel is functional 0 = Channel is faulted
ChnlFlt_RefTest_Module_A	Bit-level indicators of channels on module A that have failed the reference test. 1 = Channel faulted 0 = Channel is not faulted
ChnlFlt_RefTest_Module_B	Bit-level indicators of channels on module B that have failed the reference test. 1 = Channel faulted 0 = Channel is not faulted
Chnl_Miscompare_Status	Bit-level indicators that show what channels of the module pair do not match each other (miscompare). 1 = Channel status between modules is different 0 = Channel status is the same
ModulePair_Good	Status bit that indicates that both modules of the module pair are functioning properly. 1 = Module pair functioning properly 0 = Fault present (on one or both modules)
ModulePair_1oo1	Status bit that indicates the module pair is operating 1oo1. 1 = Operating 1oo1 0 = Either both modules of pair are OK or are faulted (that is, not in 1oo1 operation)

8 / 2011
www.klinkmann.com KLINKMANN

1756-IF16 Module Status Tags

Tag Name	Description
ModulePair_Faulted	Status bit indicates that both modules of the module pair have at least one fault. The system has failed to safe. 1 = Both modules of pair faulted 0 = Both modules of pair OK
Module_A_Faulted	Status bit indicates that module A of the pair has at least one fault. 1 = Module A faulted 0 = Module A OK
Module_B_Faulted	Status bit indicating that module B of the module pair has at least one fault 1 = Module B faulted 0 = Module B OK
Run_1oo1_Countdown	Indicates the time remaining on the 1oo1 countdown timer. The value is determined using the TimeToRun_1oo1tag value and is shown in seconds.

1756-IF16 ModulePair Tags for Use in Programming

These tags are to be used in either the main routine or in call code programs. Your program uses the data in these tags to determine system behavior.

For example, your call code routine should examine the `Run_ReferenceTest` tag. If the value of this tag is at 1, a reference test is run on the module pair.

1756-IF16 Tags for Use in Programming

Tag Name	Description
O.Data[*X*]	During normal operation, this array of channel values are the reconciled values of the two channels of the module pair. If the system is operating 1oo1, this array of channel values contains only the channel values of the unfaulted module.
IO.CircuitReset	Using programming in the Main Routine, this bit is reset manually and restarts the outputs after a fault or demand on the system.
IO.FaultReset	Using programming in the Main Routine, this bit is reset manually and resets the module status tags after a fault or demand on the system.
IO.Run_ReferenceTest	Used in the IF16_Subroutine_Call_Code, this tag value is a precondition for a DC output that is connected to the termination board of the 1756-IF16 module pair.

1756-IF16 Hidden Tags, Not for Use

Similar to the inability to access the diagnostic subroutines, there are tags within the program provided by Rockwell Automation that cannot be accessed or altered.

You cannot see these tags, however, in order to avoid potential conflicts within the program, you should not create tags with the same names.

When creating tags for your application, do not use these tags names.

1756-IF16 Tags Unavailable for Use

- ReferenceTestEn
- DataCompareTestEn
- ReferenceTestReq
- RefCalReq
- VRefs[16]
- ReferenceTestInterval
- DataCompareCounter[16]
- L_Scr[4]
- ChannelFaultsStore1
- ChannelFaultsStore2
- OneShot_Bits
- QualityMask1
- QualityMask2
- CheckforIF16ModuleFault
- FaultResetTimer
- Module_Insertion_Delay

1756-OB16D Module Pair Tags

The tags provided in the following tables are used to configure, specify, and monitor 1756-OB16D output module behavior in a ControlLogix fault-tolerant system.

1756-OB16D ModulePair Tags for System Behavior

You must enter values for each these 1756-OB16D ModulePair tags. For some tags, the value specified is required. For others, the values are recommended.

1756-OB16D ModulePair Tags Used to Specify System Behavior

Tag Name	Description	Value	Required or Recommended
I.Safety_Output_Select	Use to select or deselect the channel inputs that are used for safety functions.	1 (at each point)	Required
IO.PulseTest_Chnl_Select	Use to enable or disable the execution of pulse tests on points of the output module pair.(1) 1 = Pulse test enabled 0 = Pulse test disabled	1 (at each point)	Recommended
IO.PulseTest_Interval_PerChnl.PRE	Time, in ms, between pulse tests on individual output points. The total time it takes for pulse tests to be carried-out on all points of the module pair is this value multiplied the number of outputs. This is true even when pulse tests are disabled for any of the points. For example, when the 5 s is the PulseTest_Interval_PerChnl value, the total time required for all of the outputs to be pulse tested is 80 seconds.	5000 (5 s)	Recommended
IO.TimeToRun_1oo1.PRE	User-defined time, in ms, for the 1oo1 countdown timer that is the repair time.	28800000 (8 hours)	Recommended
IO.PulseTest_Settings[4]	Sets the maximum pulse test width and is specified in 100 µs increments.	20 (2 ms)	Required
IO.PulseTest_Settings[8]	Sets the amount of time, in 100 µs increments, for the delay between the end of the pulse test and the declaration of a fault.	20 (2 ms)	Required

(1) Pulse tests must be disabled for outputs used to trigger diagnostic tests on input module pairs and outputs used to control relays on output termination boards.

1756-OB16D Module Status Tags

The module status tags are used in several ways. Uses include:

- in the main routine to determine system behavior.
- in the subroutine to detemine and report module pair status.
- in conjunction with HMI and other indicators of system status

1756-OB16D Module Status Tags

Tag Name	Description
ConnectionFault_Module_A	Indicates the status of the connection to module A. 1 = Connection lost 0 = Connection good
ConnectionFault_Module_B	Indicates the status of the connection to module B. 1 = Connection lost 0 = Connection good
Chnl_OK_Module_A	Bit-level indicators of what points are operating without fault on module A. 1 = Point is functional 0 = Point is faulted
Chnl_OK_Module_B	Bit-level indicators of what points are operating without fault on module B. 1 = Point is functional 0 = Point is faulted
ChnlFlt_PulseTest_Module_A	Bit-level indicators of points on module A that have failed the pulse test. 1 = Point faulted 0 = Point is not faulted
ChnlFlt_PulseTest_Module_B	Bit-level indicators of points on module B that have failed the pulse test. 1 = Point faulted 0 = Point is not faulted
Chnl_Grounded_Module_A	Bit-level indicators that indicate what points are at 0, and cannot change to 1 (stuck-at-low condition). 1 = Point stuck-at-low 0 = Point able to change
Chnl_Ground_Module_B	Bit-level indicators that indicate what points are at 0, and cannot change to 1 (stuck-at-low condition). 1 = Point stuck-at-low 0 = Point able to change

1756-OB16D Module Status Tags

Tag Name	Description
Chnl_HWFail_Module_A	Status bit that indicates a hardware failure on the point of the module. 1 = Point faulted 0 = Point is not faulted
Chnl_HWFail_Module_B	Status bit that indicates a hardware failure on the point of the module. 1 = Point faulted 0 = Point is not faulted
Chnl_NoLoadOrDCV_Module_A	Indicates if the point is faulted due to a no load or DC+.(1) 1 = Point has no load 0 = Point has load
Chnl_NoLoadOrDCV_Module_B	Indicates if the point is faulted due to a no load or DC+.(1) 1 = Point has no load 0 = Point has load
O.ModulePair_Good	If both modules of the pair are functioning without faults. 1 = Both modules are functioning properly 0 = A fault is present on one or both modules of the pair
O.ModulePair_1oo1	If the module pair is operating in a 1oo1 configuration (that is, only one module of the pair is functioning properly). 1 = Module pair is operating in a 1oo1 configuration 0 = Both modules are either
O.ModulePair_Faulted	If both the modules of the pair are faulted. Depending on your application, a status of 1 at this tag may initiate a shutdown. 1 = Both modules of the pair faulted 0 = Module pair functioning properly or in a 1oo1 configuration.
O.Module_A_Faulted	The fault status of module A. 1 = Module A faulted 0 = Module A functioning properly
O.Module_B_Faulted	The fault status of module B. 1 = Module B faulted 0 = Module B functioning properly
O.Run_1oo1_Countdown	Indicates the time remaining on the 1oo1 countdown timer. The value is determined using the TimeToRun_1oo1tag value and is shown in seconds.

(1) A no load condition can be detected only if it is between the termination board and the output module.

1756-OB16D ModulePair Tags for Use in Programming

These tags are to be used in either the main routine or in call code programs. Your program uses the data in these tags to determine system behavior.

For example, your call code routine should examine the `Run_ReferenceTest` tag. If the value of this tag is at 1, a transition test is run on the module pair.

1756-OB16D Tags for Use in Programming

Tag Name	Description
IO.OneShot_Bits	This tag is used in the Subroutine_Call_Code to initiate the pulse test.
IO.PulseTestResults_Module_A	Used as a Dest parameter in MOV instructions of the Subroutine_Call_Code and is where module pulse test results are stored.
IO.PulseTestResults_Module_B	Used as a Dest parameter in MOV instructions of the Subroutine_Call_Code and is where module pulse test results are stored.
IO.CircuitReset	Using programming in the Main Routine, this bit is reset manually and restarts the outputs after a fault or demand on the system.
IO.FaultReset	Using programming in the Main Routine, this bit is reset manually and resets the module status tags after a fault or demand on the system.
IO.Run_PulseTest	This tag is examined in the OB16D_Subroutine_Call_Code and used as a precondition for the MSG instruction that initiates the Pulse Test.
Relay_Module_A	This tag is examined in the OB16D_Subroutine_Call_Code and used as a precondition for the DC output that disconnects the power (via the relay) for module A.
Relay_Module_B	This tag is examined in the OB16D_Subroutine_Call_Code and used as a precondition for the DC output that disconnects the power (via the relay) for module B.

1756-OB16D Hidden Tags, Not for Use

Similar to the inability to access the diagnostic subroutines, there are tags within the program provided by Rockwell Automation that cannot be accessed or altered.

You cannot see these tags, however, in order to avoid potential conflicts within the program, you should not create tags with the same names.

When creating tags for your application, do not use these tags names.

1756-OB16D Tags Unavailable for Use

- DataCompareTestEn
- L_Scr[4]
- OneShot_Bits
- QualityMask1
- QualityMask2
- FaultResetTimer

SIL2 Fault-tolerant Topology

About This Appendix

This appendix provides considerations for use when planning your fault-tolerant I/O system. It also includes an example layout of fault-tolerant system.

Topic	Page
Planning Considerations	149
1756-OB16D Module Pair Arrangement	151

Planning Considerations

Remember these considerations when planning and laying-out your fault-tolerant system.

Fault-tolerant System Planning Considerations

For module type	Make these considerations
1756-IB32 module pair	• Use 1492-CABLE*XXX*Z cables to connect the 1756-IB32 module pair to the input termination board . • Connect one 1756-OB16D module pair output point to the termination board wiring terminal. This output point is used to control the relay on the DC input termination board.(1) This output point, because it controls the relay on the termination board, triggers transition tests on the 1756-IB32 module pair.
1756-IF16 module pair	• Use 1492-ACABLE*XXX*UA cables to connect the 1756-IF16 module pair to the analog input termination board. • Connect one 1756-OB16D module pair output point to the termination board wiring terminal.This output point is used to control the switch on the analog input termination board.(1) This output point, because it controls the termination board switch, is used to trigger reference tests on the 1756-IF16 module pair.

Fault-tolerant System Planning Considerations

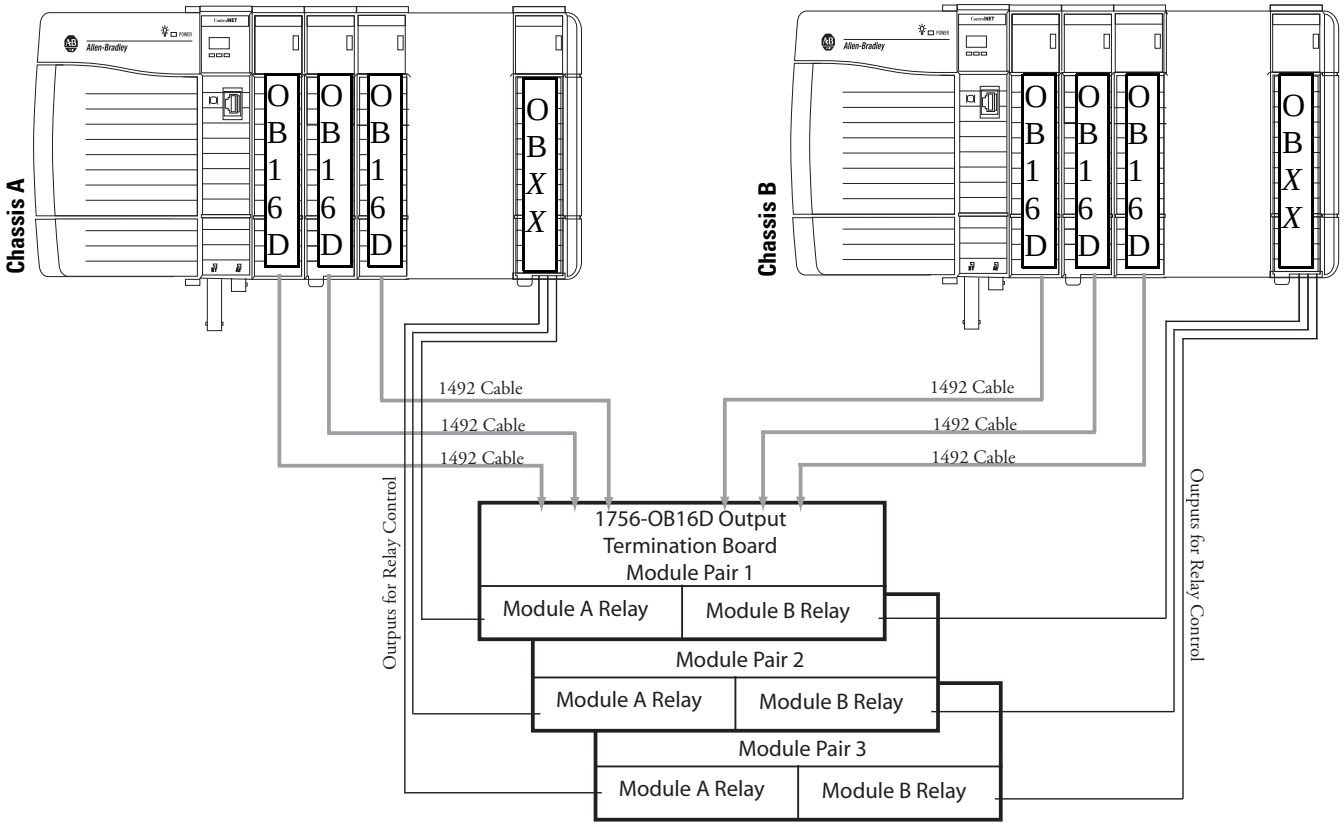
For module type	Make these considerations
1756-OB16D module pair	• Use 1492-CABLE*XXX*Z cables to connect the 1756-OB16D module pair to an output termination board. • Use two 1756-OB*XX*(2) modules to control relays on the output termination board. Connect an output from a 1756-OB*XX*(2) module to the termination board. This output point is used to control the relay for 1756-OB16D module A. Connect another 1756-OB*XX* output point to control the relay for 1756-OB16D module B. This arrangement requires that two 1756-OB*XX* output modules be used. Each 1756-OB*XX* module controls a termination board relay of a 1756-OB16D module in the module pair.(3) • Place the 1756-OB*XX* module in the *same chassis* as the 1756-OB16D module whose relay it is controlling. That is, the 1756-OB*XX* module used to control the relay for 1756-OB16D module A must be placed in chassis A of the chassis pair. The 1756-OB*XX* module used to control the relay for 1756-OB16D module B must be placed in chassis B of the chassis pair. Because the standard, 1756-OB*XX* module must be in the same chassis as the 1756-OB16D module whose relay it is controlling, consider placing all of your 1756-OB16D modules together in the same chassis in order to reduce the number of standard, 1756-OB*XX* modules required in your system.

(1) Pulse tests must be disabled on 1756-OB16D output points used to control input relays or switches.

(2) For information about which 1756-OB*XX* modules can be used to control the relays on the output module termination board, see Chapter 2, 1756-OB16D Output Termination Board Relay Control, page 42.

(3) If using 1756-OB16D modules to control the relays of your 1756-OB16D module pairs, you must disable pulse testing on the points used for relay control.

1756-OB16D Module Pair Arrangement



Fault-tolerant System Limitations

About This Appendix

This appendix describes the limitations of the fault-tolerant system.

Topic	Page
About Faults and Overall Fault-tolerance	153
Detecting System-side Versus Field-side Faults	153
Limits of Fault-detection from the 1756-OB16D Termination Board	153
Module Pair Faults	154

About Faults and Overall Fault-tolerance

The ControlLogix fault-tolerant has been designed to identify system faults, and, in most cases, continue to operate in the event of those faults. However, the fault-tolerant system does have limitations. These limitations are described in this appendix.

Detecting System-side Versus Field-side Faults

The ControlLogix fault-tolerant system can detect only system-side faults. System-side faults are those that occur within the hardware of the ControlLogix SIL2-certified fault-tolerant system.

This means that any fault that occurs beyond the fault-tolerant system hardware cannot be detected.

Limits of Fault-detection from the 1756-OB16D Termination Board

The 1756-OB16D termination board is not able to detect if a no-load condition exists on the outputs that extend from the termination board to a device.

The ControlLogix fault-tolerant system can detect a shorted wire condition between the termination board and the field device. The system is also able to detect if a wire-off condition exists between the output module and termination board.

Module Pair Faults

When certain faults occur on the fault-tolerant system, the system programming recognizes those faults as a faulted module pair - even if the fault is present only on one module of the pair. Depending on your application and main routine programming, these module pair faults may result in a system shutdown.

This table describes module pair faults that may occur in the fault tolerant system. It also describes why the fault is identified as a module pair fault that causes the system not to use data from that module pair.

Module Pair Type	Fault Type	Faulted module pair occurs because
1756-IB32	A miscompare between any two points on the module pair.	The system cannot detect a stuck-at-zero (stuck-at-low) condition. Therefore, *any* zero (low) point condition is processed as a demand on the safety system.
1756-IF16 with the use of two-sensor wiring	A miscompare between any two channels of the module pair occurs, and continues to occur, after a reference test is successfully carried-out on the module pair. The reference test indicates that the analog input modules are functioning properly. However, the miscompare of channels continues to be detected by the system after the reference test.	A hardware failure exists. The failure is likely to either be at on one of the two sensors, or, on the analog input termination board.
1756-IF16	A failure of the reference test due to incorrect reference voltages.	If the correct reference voltages are not detected, there is a fault either on the termination board or with the outputs from the 1756-OB16D module pair that trigger the reference test.
1756-OB16D	Diagnostics of the 1756-OB16D module identify a short condition in the wiring from the termination board to the load.	Because the shorted wiring is related to the output of both 1756-OB16D modules, a module pair fault occurs.
1756-IB32, 1756-IF16	Both modules of a pair fail diagnostic tests (that is, transition tests or reference tests) simultaneously.	Either: A. A hardware failure in the system caused both modules to fail the diagnostic tests. For example, if the 1756-OB16D outputs used to control the input termination board relays are damaged or the switches of the analog input termination board fail. B. Faults exist on both modules of the pair and have been identified by the diagnostic tests.
1756-IB32, 1756-IF16, and 1756-OB16D	Both modules of the pair have any type of fault or fault condition. These are example conditions. • Module A has a point fault and module B has a connection failure. • Module A has a no-load condition at one point and module B has a point with a shorted condition.	Fault conditions on both modules indicate that the system cannot safely run 1oo1 or 1oo2 and significant repairs should be made.

Frequently Asked Questions

About This Appendix

This section answers frequently asked questions specific to ControlLogix SIL2 systems and diagnostic subroutines.

Topic	Page
About Redundant Chassis	155
About I/O	157
About Fail-safe and Fault-tolerant Programs	160

About Redundant Chassis

These questions are specific to the use of redundant chassis in a SIL2 system.

Answers for each of these frequently-asked-questions are categorized based on the use of the diagnostic subroutines.

If you are	See the answers labeled
Not using the diagnostic subroutines to program your system	SIL2 General Requirements
Using the diagnostic subroutines to program your system	SIL2 Diagnostic Subroutine Requirements

Am I required to use redundant (duplicate) I/O chassis?

SIL2 General Requirements

No. If you are configuring any ControlLogix SIL2-compliant system, you do not have to configure your remote I/O into redundant (duplicate) chassis. To achieve SIL2-compliance, you may choose to use any of the hardware configurations described in the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001.

It is important to understand that your placement of I/O directly affects the availability and fault-tolerance of the SIL2 system. For an illustration of this concept, see Hardware Configurations and Fault-tolerance on page 157.

SIL2 Diagnostic Subroutine Requirements

No. You may use several different SIL2-certified configurations of your remote I/O with the diagnostic subroutines. However, the use of redundant remote-I/O chassis provides the highest level of availability compared to other SIL2 hardware configurations.

You may also choose to place I/O in non-redundant chassis remote from the controller or in the same chassis as the controller. It is important to understand that your placement of I/O directly affects the availability and fault-tolerance of the SIL2 system. For an illustration of this concept, see Hardware Configurations and Fault-tolerance on page 157.

Am I required to use redundant controller chassis?

SIL2 General Requirements

No. You may use a redundant or non-redundant controller chassis configuration for your SIL2 system. However, like the use of redundant I/O, the use of redundant controller chassis increases the availability and fault-tolerance of the SIL system.

For an illustration of this concept, see Hardware Configurations and Fault-tolerance on page 157.

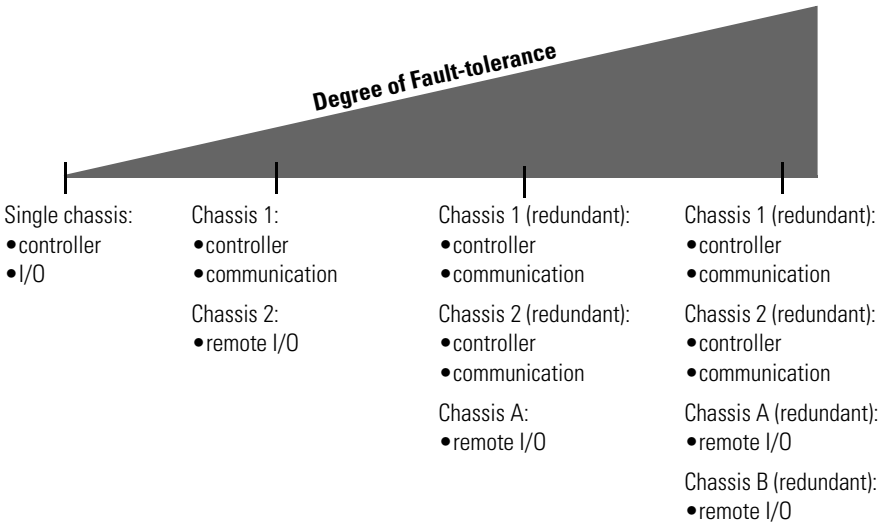
SIL2 Diagnostic Subroutine Requirements

No. The diagnostic subroutines can be used with either the redundant or non-redundant controller chassis configurations. The choice to use redundant controller and communication chassis is not affected by the use of the diagnostic subroutines because those instructions are used to program for only I/O.

More About SIL2 Hardware Configurations and Fault-tolerance

This illustration can be used as a reference when determining how to configure your SIL2 hardware to meet the requirements for your SIL2 system's fault-tolerance and availability.

Hardware Configurations and Fault-tolerance



About I/O

This sections answers frequently asked questions specific to the use of I/O modules and peripherals with the diagnostic subroutines in the SIL2 system.

Answers for each of these frequently-asked-questions are categorized based on the use of the diagnostic subroutines.

If you are	See the answers labeled
Not using the diagnostic subroutines to program your system	SIL2 General Requirements
Using the diagnostic subroutines to program your system	SIL2 Diagnostic Subroutine Requirements

Am I required to use input module pairs?

SIL2 General Requirements

Yes. If you are configuring a ControlLogix SIL2-compliant system without the diagnostic subroutines, you still have to use input module pairs. See the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001 for lists of available SIL2 hardware and usage considerations.

SIL2 Diagnostic Subroutine Requirements

Yes. If you are using the diagnostic subroutines, you are required to use input module pairs. Both the 1756-IB32 and 1756-IF16 input modules must be used as module pairs in order for the diagnostic subroutine to function as programmed.

Am I required to use 1756-OB16D module pairs?

SIL2 General Requirements

No. If you are configuring any ControlLogix SIL2-compliant system, you do not have to use 1756-OB16D module pairs. The use of module pairs is required only when your system requires the highest level of availability and fault-tolerance.

SIL2 Diagnostic Subroutine Requirements

No. The use of 1756-OB16D module pairs establishes a higher level of fault-tolerance, but is not required for the use of the diagnostic subroutines. Depending on your application, you may choose to use an independent 1756-OB16D module instead.

If you are using the diagnostic subroutines, then you must use at least one 1756-OB16D module in a manner similar to that described in this manual.

For information about editing input parameters for a single 1756-OB16D module, see this question:

- If I am configuring a fail-safe system, what parameters should I specify in the JSR for the 1756-OB16D output modules? (on page 162).

Am I required to use a standard output module to control the output relays of the 1756-OB16D termination board?

SIL2 General Requirements

Yes. If you are using the 1756-OB16D output termination boards, you must use a standard output module to control the relays of that board as described in Chapter 2 on page 38. This is because the outputs of the 1756-OB16D module cannot be used to control its own relays.

SIL2 Diagnostic Subroutine Requirements

Yes. If you are using the diagnostic subroutines, you must use a standard output module to control the relays of the 1756-OB16D termination board as described in Chapter 2 on page 38. This is because the outputs of the 1756-OB16D modules cannot be used to control their own relays.

Do I always have to use the specialized I/O termination boards?

SIL2 General Requirements

No. You are not required to use termination boards if you are not using the diagnostic subroutines.

However, if you choose not to use them, you are responsible for the comparable hardware and programming described in the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001.

SIL2 Diagnostic Subroutine Requirements

Yes. If you are using the diagnostic subroutines, you must use the specialized I/O termination boards described in Chapter 2.

Can I use I/O modules other than the 1756-IB32, 1756-IF16, and 1756-OB16D modules?

SIL2 General Requirements

Yes. If you are implementing a SIL2 system without using the diagnostic subroutines, you may use any of the I/O modules listed in the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001.

SIL2 Diagnostic Subroutine Requirements

No. If you are using the diagnostic subroutines, you can use only the I/O modules listed in Chapter 2 on page 21.

About Fail-safe and Fault-tolerant Programs

This section answers frequently asked questions specific to the programming requirements of fault-tolerant and fail-safe systems.

Unlike the previous frequently-asked-question sections, these questions are specific to the use of the diagnostic subroutines and, being so, the answers are not categorized.

Can I use the diagnostic subroutines to implement a SIL2 fail-safe system?

Yes. As long as you use the diagnostic subroutines with the required hardware, you can use the diagnostic subroutines to implement a fail-safe system.

If you use the diagnostic subroutines to implement a fail-safe system, you must adapt your program to go to the safe state in the event of a fault. For more information about programming for a fail-safe system, see the next question.

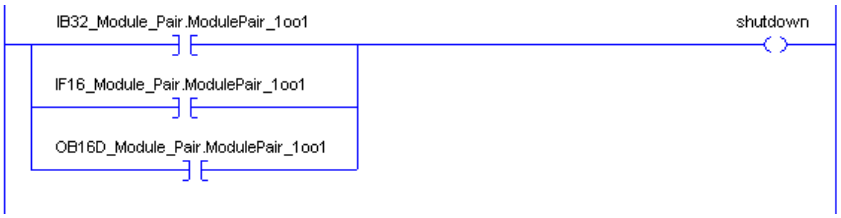
How is programming for a fail-safe system different than programming for a fault-tolerant system?

The difference between fail-safe and fault-tolerant programming is in the programmed response to a fault in the system. There are multiple possibilities for system-responses to faults that may occur.

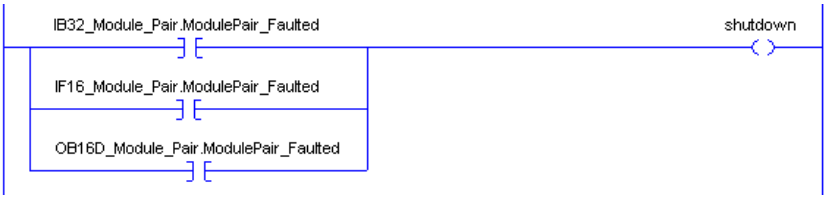
One example of a possible difference between fail-safe and fault-tolerant programming is shown in this example.

Example Fail-safe versus Fault-tolerant Program Rung

Fail-safe



Fault-tolerant



In the fail-safe rung, any faulted module results in a system shutdown - even if though the second module of the pair is still functioning properly.

As demonstrated in the fault-tolerant rung, the system shuts down only if both modules of the pair are faulted. If one module of the pair continues to function properly (that is, the module pair is operating 1oo1), the system continues to carry-out the safety function.

When programming a fail-safe system, reference the Using ControlLogix in SIL2 Applications Safety Reference Manual, publication 1756-RM001, for more fail-safe programming techniques.

If I am configuring a fail-safe system, what parameters should I specify in the SIL2 Add-On Instructions for the input module pairs?

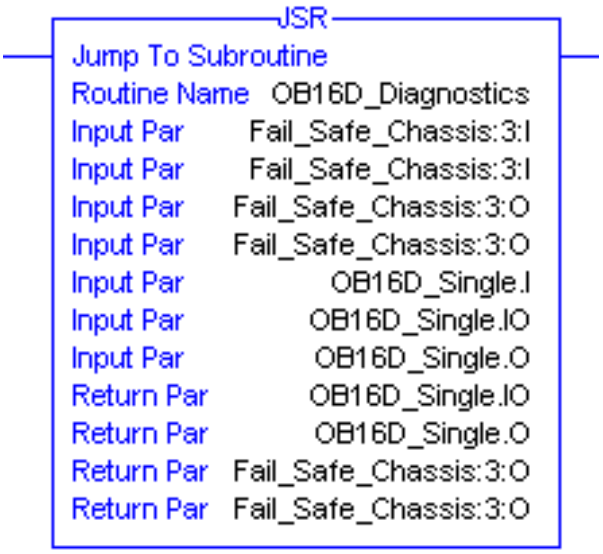
Specify the same input parameters for the input module pairs as those shown in Chapter 4 (page 57) for the fault-tolerant system.

If I am configuring a fail-safe system, what parameters should I specify in the JSR for the 1756-OB16D output modules?

- **If you are using an 1756-OB16D module pair**, specify the same parameters as those shown in Chapter 4 (page 65) for the fault-tolerant system.
- **If you are using a single 1756-OB16D module** (that is, not a module pair) with the diagnostic subroutines in a fail-safe system, the required input parameters reflect the use of only one module. For each set of input parameters that requires the use of a tag from each module of the pair, specify the same tag for the one 1756-OB16D module.

  This graphic shows an example of how the JSR is configured if only one 1756-OB16D module is used.

Parameters for 1756-OB16D Single-module Use



KLINKMANN 8 / 2011 www.klinkmann.com

These terms are used throughout this manual.

1oo1 state

Describes the state of the system when a channel, module, or chassis of a pair within the SIL2 system is faulted and the system is operating using only data from the unfaulted channels, module of the pair, or chassis of the pair.

Call_Code subroutine

A subroutine provided in the SIL2_IO_Fault_Tolerant program. It is used to call the diagnostic subroutine for each module pair.

chassis pair

A set of two remote I/O chassis used in the SIL2 fault-tolerant system. Each chassis of the pair contains a set of I/O modules that exactly match each other in both their type of modules (1756-IB32, 1756-IF16, and 1756-OB16D) and their order within the chassis.

diagnostic subroutine behavior

The manner in which the diagnostic subroutines function in the system. Behaviors of the subroutines that can be specified include: the amount of time the system operates 1oo1, the amount of time between diagnostic tests, the frequency of diagnostic tests, and the number of times a miscompare occurs before a fault is declared.

diagnostic subroutine

A subroutine provided in the SIL2_IO_Fault_Tolerant program. It carries-out a variety of tests and checks on the I/O module pairs and provides data that describes module status. The diagnostic subroutine is locked, and therefore cannot be altered.

duplicate, identical chassis pairs

A chassis pair that is configured so the type of modules (1756-IB32, 1756-IF16, and 1756-OB16D), the order of modules, and the module properties are identical between each chassis of the pair.

emergency shutdown (ESD)

When certain faults occur in the fault-tolerant SIL2 system, the inputs and outputs must be programmed to reach their safe state, which is commonly de-energized. This de-energizing is referred to as an emergency shutdown.

fail-safe configuration

A SIL2 configuration where a fault anywhere in the safety system results in a system shutdown, that is, the system fails-to-safe.

fault tolerance

The ability of a functional unit to continue to perform a required function in the presence of faults or errors. For more information, see IEC publication 61508-4.

fault-tolerant configuration

A ControlLogix system that is configured so that the system can continue to carry-out the safety function, even when certain faults occur. The fault-tolerant system is comprised of redundant controller chassis, duplicate remote I/O chassis, and I/O termination boards.

high-availability configuration

A ControlLogix system that is configured so that some types of faults can be tolerated. The high-availability configuration is comprised of redundant controller chassis and remote I/O.

module pair

A set of two I/O modules, each placed in one chassis of a chassis pair. Module pairs are I/O modules that are identical both in type (1756-IB32, 1756-IF16, or 1756-OB16D) and in their configuration within the programming software.

module pair status tags

ModulePair tags that provide the operational status of the module pair.

module status tags

ModulePair tags that provide the operational status of individual modules within the module pair.

ModulePair tags

Tags of a User-defined Data Type (UDT) created specifically for fault-tolerant, SIL2 applications. The ModulePair tags are used to specify diagnostic behavior, program system responses, and monitor the status of the I/O modules.

nonfault-tolerant SIL2-certified modules

Modules that are certified for use in SIL2 systems (for example fail-safe and high-availability) but are not certified for use in fault-tolerant systems.

normal state

Also call normal operation, this term denotes the state of the system or module when diagnostic tests are not being carried-out, nor are any of the modules faulted (for example, when the system is operating 1oo1).

recommended tag values

ModulePair tag values that Rockwell Automation provides recommended values for. However, you may choose to specify different values based upon your application.

redundant controller chassis

A set of chassis that contain controllers and communication modules that constantly check each other and function as backups for each other if a fault occurs on the controller or communication modules.

reference test

A type of diagnostic test that is run on the inputs of the 1756-IF16 analog input modules. During the reference test, reference voltages are applied to input channels and the IF16_Diagnostic subroutine verifies that the values returned by the input module match those applied (within the deadband).

required tag values

ModulePair tag values provided Rockwell Automation that must be used and are not application-dependant. Where required tag values are specified, no other values may be used.

safety integrity level (SIL)

A SIL is a level in the IEC rating system used to specify the safety integrity requirements of a safety-related control system. SIL1 is the lowest level and SIL4 is the highest. For more information about SIL specifications, see IEC publication 61508-1, General Requirements.

SIL

See safety integrity level (SIL).

stuck-at-one condition

Also called stuck-at-high, this is a condition where a digital input point cannot change from the value of 1 (or high) to 0 (low).

system-generated tags

Tags that are created by RSLogix 5000 software when you configure your I/O configuration tree.

test state

In the fault-tolerant system, this is the state where diagnostic tests (that is, transition tests or reference tests) are being carried-out and the program is operating on last-known and verified data.

transition test

A type of diagnostic test that is run on the inputs of the 1756-IB32 DC input modules. During the transition test, the termination board changes the input point values from 1 (ON) to 0 (OFF). The IB32_Diagnostics subroutine verifies that points transitioned from 1 to 0 properly.

Numerics

1756-IB32 Call_Code subroutines
edit 85–89
add JSR rung 85
edit rung elements 88
JSR parameters 87
1756-IB32 DC input termination board
features 26
figure of, normal operation 27
figure of, transition test 29
function, normal operation 27
function, transition test 28
1756-IB32 module pair
demand programming 113
diagnostic subroutines 55
identify a module fault 122
1756-IB32 ModulePair tags 131–136
editing 77
for system behavior 131
for use in programming 135
hidden 136
module status tags 133
1756-IB32 modules
properties 68
replacement 121
1756-IF16 analog input termination board
DIP switches for wiring options 33
features 30
figure of, normal operation 32
figure of, reference test 35
function, normal operation 31
function, reference tests 34
reference tests 34
two-wire transmitters with 31
wiring options 33
1756-IF16 Call_Code subroutines
edit 90–94
add JSR rung 90
edit rung elements 93
JSR parameters 92
1756-IF16 module pair
demand programming 114
diagnostic subroutines 57
identify a module fault 123
status tags 123
transmitters required 25
wiring options 33
1756-IF16 ModulePair tags 137–142
editing 79
for module status 138
for programming 141
for system behavior 137
hidden 142
1756-IF16 modules
properties 69
1756-OB16D Call_Code subroutine
edit MSG instructions 99
1756-OB16D Call_Code subroutines
edit 95–103
add JSR rung 95
rung elements 97
1756-OB16D diagnostic output termination board
diagnostic tests and 39
features 37
function during normal operation 38
1756-OB16D module pair
diagnostic subroutines 60
status tags 124
1756-OB16D module pair chassis
example of 151
1756-OB16D ModulePair tags 143–147
editing 82
for module status 144
for programming 146
for system behavior 143
hidden 147
1756-OB16D modules
properties 70
1756-OB16D outputs
used to control input diagnostic tests 44
1oo1
state 53

C

Call_Code subroutines
edit the 1756-IB32 85–89
add JSR rung 85
edit rung elements 88
JSR parameters 87
edit the 1756-IF16 90–94
add JSR rung 90
edit rung elements 93
JSR parameters 92
edit the 1756-OB16D 95–103
add JSR rung 95
edit rung elements 97
editing 84–103
element in the fault-tolerant program 49
channel comparision
deadbands in normal operation 80
channel voltages, reference test 36
channel-level programming 106
chassis pair
output module chassis 151

chassis pairs
identical duplicates 17
in fault-tolerant configurations 16
limits 16
naming conventions 68
termination board use with 17
circuit reset 111
when to use 125
CNBR, add to program 66
configurations
fail safe 14
fault-tolerant 15, 16
high-availability 14
SIL2-certified 13
configuring the system 65–103
add a CNBR 66
add the remote I/O chassis 67
configure the remote I/O chassis 67
configure the remote I/O modules 67
prepare redundant controller chassis 65
resulting I/O configuration tree 70
resulting system-generated tags 71
specify I/O module properties 68
start with program 66
considerations for planning 149
controller chassis 156
ControlLogix
fault tolerance 14
SIL2 configurations 13

D

data
.I and .O in the program 106
flow in program 62
use in program 106
deadbands
channel comparision 80
for reference tests 36
demand programming 113
for 1756-IB32 module pair 113
for 1756-IF16 module pair 114
diagnostic subroutines
element in the fault-tolerant program 48, 50
features of 48
IB32_Diagnostics subroutine 55
1oo1 56
normal operation 55
test 56
IF16_Diagnostics subroutine 57
1oo1 58
normal operation 57
test 58
main routine and 106
OB16D_Diagnostics subroutine 60
1oo1 61
normal operation 60
diagnostic tests
1756-IB32 module pair 28
1756-IF16 module pair 34
1756-OB16D module pair 39
control of 44
reference tests 34
transition tests 28
DIP switches, on analog termination board 33

E

Edit 97
elements of the fault-tolerant program
Call_Code subroutines 49
data flow between 62
diagnostic subroutines 48
figure of in software 51
functions 50
main routine 47

F

fail-safe
diagnostic subroutines and 160
programming 161
fail-safe configuration
about 14
fault programming
circuit reset 111
module pair 108
reset fault 109
fault reset 109
when to use 125
fault tolerance
ControlLogix system and 14
fault tolerance and ControlLogix 13–21
faulted module pair
example programming to identify 120
tags to identify 118
faulted state 54
faults
cause of input diagnostic test failures 44

fault-tolerant
about 14
configuration 15
configuration compared to others 15
configuration description 16
program
elements 47–51
fault-tolerant program
start configuration 66
fault-tolerant system
I/O modules for use in 25
planning considerations 149
termination boards for use in 25
fault-tolerant system, configuring 65–103
add a CNBR 66
add remote I/O chassis 67
prepare redundant controller chassis 65
remote I/O chassis 67
remote I/O modules 67
specify I/O module properties 68
start with program 66

H

hardware
configurations and fault-tolerance 157
I/O chassis configurations 155
high-availability configuration
about 14
figure of 15

I

I/O configuration tree
after configuration 70
I/O in fault-tolerant configurations 16
I/O module
faults, use of reset to clear 125
programming to identify faulted 121
I/O module properties, specify 68
I/O modules
approved for fault-tolerant system 25
input
required 158
output
required 158
standard I/O 160
standard output
required 159
termination boards functions 18
IB32_Diagnostics subroutine
1oo1 56
about 55
normal operation 55
test 56
identical, duplicate remote I/O chassis
about 17
figure of 17
required 155
IF16_Diagnostics subroutine
1oo1 58
about 57
normal operation 57
test 58
IF16_RefCal
purpose of 59
input termination board
function during transition test 28, 35
input/output programming 106

J

JSR parameters
for 1756-IB32 module pair 87
for 1756-IF16 module pair 92

L

limits on chassis pairs 16

M

main routine
data use in 106
diagnostic subroutines and 106
element in the fault-tolerant program 47
programming 105–115
MESSAGE tags
add to the program 84
use in 1756-OB16D Call_Code 99
module pairs
example programming to identify faulted 120
fault programming 108
identify faulted 118
use resets to clear faults 125
module properties
1756-IB32 modules 68
1756-IF16 modules 69
1756-OB16D modules 70
specify in program 68
module status tags
listed 119

module tags 71
ModulePair tags
1756-IF16
module status 123
1756-OB16D module status 124
about 72
edit 76–83
editing
1756-IB32 tags 77
1756-IF16 tags 79
1756-OB16D tags 82
example, 1756-IF16 fault values 127–128
for 1756-IB32 131–136
for programming 135
hidden 136
module status tags 133
system behavior 131
for 1756-IF16 137–142
for module status 138
for programming 141
hidden 142
system behavior 137
for 1756-OB16D 143–147
for module status 144
for programming 146
for system behavior 143
hidden 147
for module status 119
naming conventions 73
to identify faulted 1756-IB32 modules 122
to identify faulted 1756-IF16 modules 123
to identify faulted module pair 118
to identify faulted modules 121
modules, identify faulted 121
MSG instruction
edit in 1756-OB16D Call_Code 99
MSG instructions
properties for 100

N

naming conventions
chassis pair and modules 68
ModulePair tags 73
normal state 52

O

OB16D_Diagnostics subroutine
1oo1 61
about 60
normal operation 38, 60
one-sensor wiring 33
output module pair
chassis configuration 151
outputs and diagnostic tests 44

P

planning considerations 149
point-level programming 106
program
elements
figure of in software 51
program elements 47–63
Call_Code subroutines 49
data flow between 62
diagnostic subroutines 48
functions 50
main routine 47
program the main routine 105–116
programming
circuit reset 111
example to identify faulted module pair 120
fault reset 109
for demand 113
on 1756-IB32 module pair 113
on 1756-IF16 module pair 114
for module pair 108
software requirements 21
to identify faulted modules 121
use of .I and .O data 106
programming the main routine 105–115

R

reconciled input data 107
redundant controller chassis
configure in fault-tolerant program 65
required 156
reference test
calibration logic 59
reference tests 34–36
analog termination board and 34
channel voltages applied 36
deadbands for 36
figure of analog input termination board during 35
purpose 34

remote I/O modules
add to the program 67
approved modules 25
chassis configuration 16
configure in program 67
termination boards and 18
remote I/O modules, configure in the program 67
replace
faulted 1756-IB32 module 121
resets
use of after faults 125

S

SIL
about 11
explanation of levels 11
SIL2 configurations, ControlLogix 13
software
requirements 21
states
1oo1 53
faulted 54
normal 52
test 52
subroutines
Call_Code
about 49
editing 84
diagnostic
about 48
IF16_RefCal 59
system states 52–54
system-generated tags 71

T

tags
1756-IF16 module status 123
1756-OB16D module status 124
create ModulePair 73
edit ModulePair 76
fault reset programming 110
MESSAGE
add 84
use in 1756-OB16D Call_Code 99
module status 119
ModulePair 72
edit for 1756-IB32 77
edit for 1756-IF16 79
edit for 1756-OB16D 82
used to identify faulted modules 121
ModulePair, create 73
system-generated 71
used to identify faulted module pair 118
user-defined data types 72
termination boards
about 26
and I/O modules 25
approved 25
I/O-specific functions 18
interaction with I/O 18
relay control 40–43
input termination board relay control 40
output termination board relay control 41
required 159
used with chassis pairs 17
test state 52
The 30
transition tests
1756-OB16D outputs and 28
about 28
figure of termination board during 29
function of termination board during 28
intervals between 28, 34, 35
purpose 28
termination board during 28, 35
transmitters
for use with 1756-IF16 module pair 25
troubleshooting
identify faulted module pair 118
identify faulted modules 121
troubleshooting a system 117–128
two-sensor wiring 33
two-wire transmitters, use with 1756-IF16 modules 31

U

user-defined data types
create ModulePair tags 73
ModulePair tags 72

Publication 1756-AT010B-EN-P - October 2008

Rockwell_Software_RSLogix-5000_ControlLogix_SIL2_System_en_0811.pdf

KLINKMANN
www.klinkmann.com

Helsinki
tel. +358 9 540 4940
automation@klinkmann.fi

St. Petersburg
tel. +7 812 327 3752
klinkmann@klinkmann.spb.ru

Moscow
tel. +7 495 641 1616
moscow@klinkmann.spb.ru

Yekaterinburg
tel. +7 343 376 5393
yekaterinburg@klinkmann.spb.ru

Samara
tel. +7 846 273 95 85
samara@klinkmann.spb.ru

Kiev
tel. +38 044 495 33 40
klinkmann@klinkmann.kiev.ua

Riga
tel. +371 6738 1617
klinkmann@klinkmann.lv

Vilnius
tel. +370 5 215 1646
post@klinkmann.lt

Tallinn
tel. +372 668 4500
klinkmann.est@klinkmann.ee

Minsk
tel. +375 17 200 0876
minsk@klinkmann.com